Package ‘PostcodesioR’

December 1, 2021

Type Package

Title API Wrapper Around 'Postcodes.io'

Version 0.3.1

Maintainer Eryk Walczak <erykjwalczak@gmail.com>

Description Free UK geocoding using data from Office for National Statistics. It is using several functions to get information about post codes, outward codes, reverse geocoding, nearest post codes/outward codes, validation, or randomly generate a post code. API wrapper around <https://postcodes.io>.

License GPL-3

URL https://docs.ropensci.org/PostcodesioR/

Depends R (>= 3.1)

Imports httr

RoxygenNote 7.1.2

Suggests knitr, dplyr, rmarkdown, testthat, covr, purrr

VignetteBuilder knitr

BugReports https://github.com/ropensci/PostcodesioR/issues

NeedsCompilation no

Author Eryk Walczak [aut, cre] (<https://orcid.org/0000-0001-8724-462X>), Claudia Vitolo [rev] (<https://orcid.org/0000-0002-4252-1176>), Robin Lovelace [rev] (<https://orcid.org/0000-0001-5679-6536>)

Repository CRAN

Date/Publication 2021-12-01 00:40:11 UTC

R topics documented:

  bulk_postcode_lookup .............................................. 2
  bulk_reverse_geocoding ........................................... 3
  nearest_outcode .................................................. 4
  nearest_outcode_lonlat .......................................... 5
  nearest_postcode .................................................. 5
bulk_postcode_lookup

Description
Returns a list of matching postcodes and respective available data.

Usage
bulk_postcode_lookup(postcodes)

Arguments
postcodes Accepts a list of postcodes. Accepts up to 100 postcodes. For only one postcode use postcode_lookup.

Value
A list of length one.

See Also
postcode_lookup for documentation.

Examples

pc_list <- list(
    postcodes = c("PR3 0SG", "M45 6GN", "EX165BL")) # spaces are ignored
bulk_postcode_lookup(pc_list)
# The function needs a list of length one. This won't work:
bulk_postcode_lookup(list("PR3 0SG", "M45 6GN", "EX165BL"))
bulk_reverse_geocoding

---

**Description**

Returns nearest postcodes for a given longitude and latitude. Accepts up to 100 geolocations.

**Usage**

`bulk_reverse_geocoding(geolocations)`

**Arguments**

- `geolocations`: A list containing an array of objects to geolocate. At least two elements needed.

**Details**

This method requires a JSON object containing an array of geolocation objects to be POSTed. Each geolocation object accepts an optional radius (meters) and limit argument. Both default to 100m and 10 respectively. It also accepts a wideSearch argument.

**Value**

A list with the geocoded data.

**See Also**

- `postcode_lookup` for documentation.

**Examples**

```r
geolocations_list <- structure(
  list(
    geolocations = structure(
      list(
        longitude = c(-3.15807731271522, -1.12935802905177),
        latitude = c(51.4799900627036, 50.7186356978817),
        limit = c(NA, 100L),
        radius = c(NA, 500L)),
      .Names = c("longitude", "latitude", "limit", "radius"),
      class = "data.frame",
      row.names = 1:2)),
    .Names = "geolocations")

bulk_reverse_geocoding(geolocations_list)
```
nearest_outcode  Nearest outcode

Description

Returns nearest outcodes for a given outcode. The search is based on the relative distance of the outcode centroid.

Usage

nearest_outcode(outcode, limit = 10, radius = 5000)

Arguments

outcode A string with a UK postcode.
limit An integer. Optional parameter. Limits number of postcodes matches to return. Defaults to 10. Needs to be less than 100.
radius An integer. Optional parameter. Limits number of postcodes matches to return. Defaults to 5,000m. Needs to be less than 25,000m.

Value

A list of geographical properties.

See Also

postcode_lookup for documentation.

Examples

nearest_outcode("EC1Y")
nearest_outcode("EC1Y", limit = 11)
nearest_outcode("EC1Y", limit = 11, radius = 6000)
nearest_outcode_lonlat

Description

Returns nearest outward codes for a given longitude and latitude. The search is based on the relative distance of the outcode centroid.

Usage

nearest_outcode_lonlat(longitude, latitude)

Arguments

longitude  A string or numeric. Needs to have at least three decimal points.
latitude   A string or numeric. Needs to have at least three decimal points.

Value

A list with available data.

See Also

postcode_lookup for documentation.

Examples

nearest_outcode_lonlat(0.127, 51.507)

nearest_postcode  Nearest postcode

Description

Returns nearest postcodes for a given postcode. The search is based on the relative distance of the postcode centroid.

Usage

nearest_postcode(postcode, limit = 10, radius = 100)
Arguments

postcode A string. Valid UK postcode.
limit A string or integer. Limits number of postcodes matches to return. Defaults to 10. Needs to be lower than 100.
radius Limits number of postcodes matches to return. Defaults to 100m. Needs to be less than 2,000m.

Value

A list of geographic properties of the nearest postcode.

Examples

nearest_postcode("EC1Y 8LX")
nearest_postcode("EC1Y 8LX", limit = 11)
nearest_postcode("EC1Y 8LX", limit = 12, radius = 200)

outcode_reverse_geocoding

Outcode reverse geocoding

Description

Returns nearest outcodes for a given longitude and latitude.

Usage

code_reverse_geocoding(longitude, latitude, limit = 10, radius = 5000)

Arguments

longitude A string, integer or float. Needs to have at least two decimal points.
latitude A string, integer or float. Needs to have at least two decimal points.
limit A string, integer or float. Limits number of postcodes matches to return. Defaults to 10. Needs to be less than 100.
radius A string, integer or float. Limits number of postcodes matches to return. Defaults to 5,000m. Needs to be less than 25,000m.

Value

A list of geographical properties.
outward_code_lookup

See Also

postcode_lookup for documentation.

Examples

```r
outcode_reverse_geocoding("-3.15", "51.47")
outcode_reverse_geocoding(-3.15, 51.47)
outcode_reverse_geocoding("-3.15807731271522", "51.4799900627036")
outcode_reverse_geocoding(-3.15, 51.47, limit = 11, radius = 20000)
```

outward_code_lookup  Outward code lookup

Description

Geolocation data for the centroid of the outward code specified.

Usage

```r
outward_code_lookup(outcode)
```

Arguments

- **outcode**: A string. The outward code representing the first half of any postcode (separated by a space).

Value

The list of geographical properties.

See Also

postcode_lookup for documentation.

Examples

```r
outward_code_lookup("E1")
```
Description

Lookup a place by code. Returns all available data if found. Returns 404 if a place does not exist.

Usage

place_lookup(code)

Arguments

code A string. The unique identifier for places - Ordnance Survey (OSGB) code.

Value

A list with available places.

- code A unique identifier that enables records to be identified easily. The identifier will be persistent for all LocalTypes except Section of Named Road and Section of Numbered Road.
- name_1 Name. The proper noun that applies to the real world entity. Names that are prefixed by the definite article are not formatted for alphabetical sorting, that is, 'The Pennines' not 'Pennines, The'.
- name_1_lang Language of Name. The language type is only set where more than one name exists E.g. cym (Welsh), eng (English), gla (Scottish Gaelic).
- name_2 Name (in case of multiple languages). The proper noun that applies to the real world entity. Names that are prefixed by the definite article are not formatted for alphabetical sorting, that is, 'The Pennines' not 'Pennines, The'.
- name_2_lang Language of Name. The language type is only set where more than one name exists E.g. cym (Welsh), eng (English), gla (Scottish Gaelic).
- local_type The Ordnance Survey classification for the named place being represented by the specific feature. E.g. City, Town, Village, Hamlet, Other Settlement, Suburban Area
- outcode The postcode district, for example, SO15.
- county_unitary Administrative Area. The name of the County (non-metropolitan or Metropolitan), Unitary Authority or Greater London Authority administrative area that the point geometry for feature is within or nearest to.
- county_unitary_type Administrative Area Type. Classifies the type of administrative unit.
- district_borough District or Borough. The name of the District, Metropolitan District or London Borough administrative unit that the point geometry for the feature is within.
- district_borough_type Borough Type. Classifies the type of administrative unit.
- region The name of the European Region (was Government O ice Region) that the point geometry for the feature is within or nearest to.
- country The country (i.e. one of the four constituent countries of the United Kingdom or the Channel Islands or the Isle of Man) to which each place is assigned.
- longitude The WGS84 longitude given the Place’s national grid reference.
- latitude The WGS84 latitude given the Place’s national grid reference.
- eastings The Ordnance Survey postcode grid reference Easting to 1 metre resolution; blank for postcodes in the Channel Islands and the Isle of Man.
- northings The Ordnance Survey postcode grid reference Northing to 1 metre resolution; blank for postcodes in the Channel Islands and the Isle of Man.
- min/max northings/eastings Minimum and Maximum, Northings and Eastings. Bounding box or Minimum Bounding Rectangle (MBR) for roads and settlements. For Settlements and Sections of Named and Numbered Roads, the MBR gives a representation of the extent of these features and is not snapped to the real world extent.

See https://postcodes.io/docs for more details.

Examples

```r
place_lookup("osgb400000007454700")
```

---

**Description**

Submit a place query and receive a complete list of places matches and associated data. This function is similar to `place_lookup` but it returns a list and allows limiting the results.

**Usage**

```r
place_query(place, limit = 10)
```

**Arguments**

- `place` A string. Name of a place to search for.
- `limit` An integer. Limits the number of matches to return. Defaults to 10. Needs to be less than 100.

**Value**

A list with available places.

**See Also**

`place_lookup` for documentation.
Examples

```r
place_query("Hills")
place_query("Hills", limit = 12)
```

**Description**

Returns a data frame of matching postcodes.

**Usage**

```r
postcode_autocomplete(postcode, limit = 10)
```

**Arguments**

- **postcode**: A string. Valid UK postcode.
- **limit**: An integer. Limits number of postcodes matches to return. Defaults to 10. Needs to be less than 100.

**Value**

A data frame with suggested postcodes.

**Examples**

```r
postcode_autocomplete("E1")
postcode_autocomplete("E1", limit = 11)
```
Description

Lookup a postcode.

Usage

postcode_lookup(postcode)

Arguments

postcode A string. One valid UK postcode. This function is case- and space-insensitive. For more than one postcode use bulk_postcode_lookup. For Scottish postcodes use scottish_postcode_lookup.

Value

A data frame. Returns all available data if found. Returns NAs if postcode does not exist (404).

- postcode Postcode. All current ('live') postcodes within the United Kingdom, the Channel Islands and the Isle of Man, received monthly from Royal Mail. 2, 3 or 4-character outward code, single space and 3-character inward code.
  - 1 - within the building of the matched address closest to the postcode mean
  - 2 - as for status value 1, except by visual inspection of Landline maps (Scotland only)
  - 3 - approximate to within 50m
  - 4 - postcode unit mean (mean of matched addresses with the same postcode, but not snapped to a building)
  - 5 - imputed by ONS, by reference to surrounding postcode grid references
  - 6 - postcode sector mean, (mainly PO Boxes)
  - 8 - postcode terminated prior to Gridlink(R) initiative, last known ONS postcode grid reference
  - 9 - no grid reference available
- eastings Eastings. The Ordnance Survey postcode grid reference Easting to 1 metre resolution; blank for postcodes in the Channel Islands and the Isle of Man. Grid references for postcodes in Northern Ireland relate to the Irish Grid system.
- northings Northings. The Ordnance Survey postcode grid reference Easting to 1 metre resolution; blank for postcodes in the Channel Islands and the Isle of Man. Grid references for postcodes in Northern Ireland relate to the Irish Grid system.
- country Country. The country (i.e. one of the four constituent countries of the United Kingdom or the Channel Islands or the Isle of Man) to which each postcode is assigned.
- nhs_ha Strategic Health Authority. The health area code for the postcode.
- **longitude** Longitude. The WGS84 longitude given the Postcode’s national grid reference.
- **latitude** Latitude. The WGS84 latitude given the Postcode’s national grid reference.
- **european_electoral_region** European Electoral Region (EER). The European Electoral Region code for each postcode.
- **primary_care_trust** Primary Care Trust (PCT). The code for the Primary Care areas in England, LHBs in Wales, CHPs in Scotland, LCG in Northern Ireland and PHD in the Isle of Man; there are no equivalent areas in the Channel Islands. Care Trust/ Care Trust Plus (CT) / local health board (LHB) / community health partnership (CHP) / local commissioning group (LCG) / primary healthcare directorate (PHD).
- **region** Region (formerly GOR). The Region code for each postcode. The nine GORs were abolished on 1 April 2011 and are now known as ‘Regions’. They were the primary statistical subdivisions of England and also the areas in which the Government Offices for the Regions fulfilled their role. Each GOR covered a number of local authorities.
- **lsoa** 2011 Census lower layer super output area (LSOA). The 2011 Census lower layer SOA code for England and Wales, SOA code for Northern Ireland and data zone code for Scotland.
- **msoa** 2011 Census middle layer super output area (MSOA). The 2011 Census middle layer SOA (MSOA) code for England and Wales and intermediate zone for Scotland.
- **incode** Incode. 3-character inward code that is following the space in the full postcode.
- **outcode** Outcode. 2, 3 or 4-character outward code. The part of postcode before the space.
- **parliamentary_constituency** Westminster Parliamentary Constituency. The Westminster Parliamentary Constituency code for each postcode.
- **admin_district** District. The current district/unitary authority to which the postcode has been assigned.
- **parish** Parish (England)/ community (Wales). The smallest type of administrative area in England is the parish (also known as 'civil parish'); the equivalent units in Wales are communities.
- **admin_county** County. The current county to which the postcode has been assigned.
- **admin_ward** Ward. The current administrative/electoral area to which the postcode has been assigned.
- **ccg** Clinical Commissioning Group. Clinical commissioning groups (CCGs) are NHS organisations set up by the Health and Social Care Act 2012 to organise the delivery of NHS services in England.
- **nuts** Nomenclature of Units for Territorial Statistics (NUTS) / Local Administrative Units (LAU) areas. The LAU2 code for each postcode. NUTS is a hierarchical classification of spatial units that provides a breakdown of the European Union’s territory for producing regional statistics which are comparable across the Union. The NUTS area classification in the United Kingdom comprises current national administrative and electoral areas, except in Scotland where some NUTS areas comprise whole and/or part Local Enterprise Regions. NUTS levels 1-3 are frozen for a minimum of three years and NUTS levels 4 and 5 are now Local Administrative Units (LAU) levels 1 and 2 respectively.
- **_code** Returns an ID or Code associated with the postcode. Typically these are a 9 character code known as an ONS Code or GSS Code. This is currently only available for districts, parishes, counties, CCGs, NUTS and wards.

See `https://postcodes.io/docs` for more details.
Examples

postcode_lookup("EC1Y8LX")
postcode_lookup("EC1Y 8LX")  # spaces are ignored
postcode_lookup("DE3 5LF")  # terminated postcode returns NAs

Description

Submit a postcode query and receive a complete list of postcode matches and all associated postcode data.

Usage

postcode_query(postcode, limit = 10)

Arguments

- **postcode**: A string. Valid UK postcode.
- **limit**: An integer. Limits the number of matches to return. Defaults to 10. Needs to be less than 100.

Value

A list of geographic properties. To return a data frame use `postcode_lookup`.

Examples

postcode_query("EC1Y8LX")
postcode_query("EC1", limit = 11)
**postcode_validation**  
*Postcode validation*

**Description**  
Convenience method to validate a postcode.

**Usage**  
`postcode_validation(postcode)`

**Arguments**  
- *postcode*  
  A string. Valid UK postcode.

**Value**  
A logical vector: True or False (meaning respectively valid or invalid postcode).

**Examples**  
```r  
postcode_validation("EC1Y 8LX") # returns TRUE  
postcode_validation("XYZ") # returns FALSE  
```

---

**random_place**  
*Random place*

**Description**  
Returns a random place and all associated data

**Usage**  
`random_place()`

**Value**  
A data frame describing a random place and all associated data.

**See Also**  
`place_lookup` for documentation.
random_postcode

Examples

```r
random_place()
```

<table>
<thead>
<tr>
<th>random_postcode</th>
<th>Random postcode</th>
</tr>
</thead>
</table>

**Description**

Returns a random postcode and all available data for that postcode.

**Usage**

```r
random_postcode(outcode = NULL)
```

**Arguments**


**Value**

A list with a random postcode with corresponding characteristics.

**See Also**

`postcode_lookup` for documentation.

**Examples**

```r
random_postcode()
random_postcode("N1")
```
reverse_geocoding

Reverse geocoding

Description

Returns nearest postcodes for a given longitude and latitude.

Usage

reverse_geocoding(
    longitude,  # A string or numeric. Needs to have at least three decimal points.
    latitude,   # A string or numeric. Needs to have at least three decimal points.
    limit = 10, # An integer. Limits number of postcodes matches to return. Defaults to 10.
    radius = 100,  # Needs to be less than 100.
    wideSearch = NULL  # An integer. Limits number of postcodes matches to return. Defaults to 100m.
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>longitude</td>
<td>A string or numeric. Needs to have at least three decimal points.</td>
</tr>
<tr>
<td>latitude</td>
<td>A string or numeric. Needs to have at least three decimal points.</td>
</tr>
<tr>
<td>limit</td>
<td>An integer. Limits number of postcodes matches to return. Defaults to 10. Needs to be less than 100.</td>
</tr>
<tr>
<td>radius</td>
<td>An integer. Limits number of postcodes matches to return. Defaults to 100m. Needs to be less than 2,000m.</td>
</tr>
<tr>
<td>wideSearch</td>
<td>TRUE or FALSE. Search up to 20km radius, but subject to a maximum of 10 results. Since lookups over a wide area can be very expensive, we've created this method to allow you choose to make the trade off between search radius and number of results. Defaults to false. When enabled, radius and limits over 10 are ignored.</td>
</tr>
</tbody>
</table>

Value

A list with available data.

See Also

postcode_lookup for documentation.

Examples

reverse_geocoding(0.127, 51.507)
reverse_geocoding("0.1275", "51.5073", limit = 3)
reverse_geocoding("0.1275", "51.5073", limit = 11, radius = 200)
**Description**

Lookup a Scottish postcode.

**Usage**

```
scottish_postcode_lookup(postcode)
```

**Arguments**

- **postcode**
  A string. One valid Scottish postcode. This function is case- and space-insensitive. For non-Scottish postcodes use `postcode_lookup`. For more than one non-Scottish postcode use `bulk_postcode_lookup`.

**Value**

A data frame. Returns all available data if found. Returns NAs if postcode does not exist (404).

- postcode Postcode. Royal Mail postcode.

See [https://postcodes.io/docs](https://postcodes.io/docs) for more details.

**Examples**

```
scottish_postcode_lookup("EH12NG")
```

---

**Description**

Returns month and year if a postcode was terminated or is no longer active.

**Usage**

```
terminated_postcode(postcode)
```
Arguments

code A string. Terminated UK postcode.

Value

A data frame with data about terminated postcode. NULL if postcode is active.

- postcode Postcode. All currently terminated postcodes within the United Kingdom, the Channel Islands and the Isle of Man, received every 3 months from Royal Mail. 2, 3 or 4-character outward code, single space and 3-character inward code.
- year_terminated Termination year. Year of termination of a postcode.
- month_terminated Termination month. Month of termination of a postcode. 1-January, 2-February, ..., 12-December.
- longitude Longitude. The WGS84 longitude given the Postcode's national grid reference.
- latitude Latitude. The WGS84 latitude given the Postcode's national grid reference.

See https://postcodes.io/docs for more details.

Examples

terminated_postcode("EC1Y 8LX") # existing postcode
terminated_postcode("E1W 1UU") # terminated postcode
Index

bulk_postcode_lookup, 2, 11, 17
bulk_reverse_geocoding, 3

nearest_outcode, 4
nearest_outcode_lonlat, 5
nearest_postcode, 5

outcode_reverse_geocoding, 6
outward_code_lookup, 7

place_lookup, 8, 9, 14
place_query, 9
postcode_autocomplete, 10
postcode_lookup, 2–5, 7, 11, 13, 15–17
postcode_query, 13
postcode_validation, 14

random_place, 14
random_postcode, 15
reverse_geocoding, 16

scottish_postcode_lookup, 11, 17

terminated_postcode, 17