Package ‘ukpolice’
August 3, 2020

Title Download Data on UK Police and Crime
Version 0.2.2
Description Downloads data from the 'UK Police' public data API, the full docs of which are available at <https://data.police.uk/docs/>. Includes data on police forces and police force areas, crime reports, and the use of stop-and-search powers.


BugReports https://github.com/EvanOdell/ukpolice/issues
License MIT + file LICENSE
Imports jsonlite, tibble, purrr, httr, snakecase
Encoding UTF-8
LazyData true
RoxygenNote 7.1.0
Suggests testthat, covr, knitr, rmarkdown, ggplot2, dplyr, leaflet, htmltools, scales, sf, sp
VignetteBuilder knitr
NeedsCompilation no
Author Evan Odell [aut, cre] (<https://orcid.org/0000-0003-1845-808X>), Tierney Nicholas [aut] (<https://orcid.org/0000-0003-1460-8722>)
Maintainer Evan Odell <evanodell191@gmail.com>
Repository CRAN
Date/Publication 2020-08-03 10:10:02 UTC

R topics documented:

ukpolice-package ................................................................. 2
ukc_available ................................................................. 2
ukc_crime_category .......................................................... 3
uk_police-package

ukpolice: Download Data on UK Police and Crime

Description

Downloads data from the 'UK Police' public data API. Includes data on UK police forces and police force areas, crime reports, and the use of stop-and-search powers.

Details

Most functions return a tibble with the requested data, with the exception of `ukc_specific_outcome()` in some cases, which returns a list containing a list and a tibble.

Data accessed through ukpolice is licenced with Open Government Licence v3.0.

The API allows for 15 requests each second, but up to 30 in a single second if in a single burst. The API does not require authentication. See the API documentation for more details.

ukc_available

Available data

Description

Returns a tibble with all available datasets. The id column contains the year and month, and other columns contain a list with all police forces reporting data for that month. As of 2019-04-02 only information on stop and search data is returned by this endpoint.

Usage

ukc_available()
**ukc_crime_category**

*Crime categories*

**Description**

Crime categories

**Usage**

ukc_crime_category()

**Value**

A tibble with all available categories of crime.

---

**ukc_crime_loc**

*Crimes at a specific location*

**Description**

Returns details of crimes within a one mile radius of a given point, at a specific location ID, or from within a custom polygon.

**Usage**

ukc_crime_loc(location, date = NULL, ...)

ukc_crime_coord(lat, lng, date = NULL, ...)

ukc_crime_poly(poly_df, date = NULL, ...)

ukc_crime_location(lat, lng, location, date = NULL, ...)

**Arguments**

- **location**
  - If specified, lat and lng are ignored. Location IDs are available through other methods including `ukc_street_crime()`.

- **date**
  - The year and month in "YYYY-MM" form. If NULL, latest available month will be returned. Also accepts dates in formats that can be coerced to Date class with `as.Date()`.

- **...**
  - Further arguments passed to `httr::GET`.

- **lat**
  - Latitude. Accepts a single value.

- **lng**
  - Longitude. Accepts a single value.
poly_df: a dataframe containing the lat/ lng pairs which define the boundary of the custom area, or a sf or SpatialPointsDataFrame object. The first and last coordinates need not be the same — they will be joined by a straight line once the request is made. If a dataframe, the lat/ lng must be the first two columns, or named "lat" and "lng".

Details

If specified, lat and lng must be the same length. location or both lat and lng must be specified.

Value

A tibble with details of crimes at a given location.

Note

The API will return a 400 status code in response to a GET request longer than 4094 characters.

Examples

```r
## Not run:
x <- ukc_crime_location(lat = 52, lng = 0)
y <- ukc_crime_location(location = 802171)
poly_df_3 <- data.frame(
  lat = c(52.268, 52.794, 52.130),
  lng = c(0.543, 0.238, 0.478)
)
z <- ukc_crime_poly(poly_df_3)
## End(Not run)
```

ukc_crime_no_location

Crimes without location

Description

Returns details of crimes that cannot be mapped to a particular location. Note that the police force must be specified.

Usage

```r
ukc_crime_no_location(force, crime_category = NULL, date = NULL)
```
Arguments

force  A string containing the name of the police force to return data for. Must be specified, and is not case sensitive.

crime_category  The category of crime to return. Defaults to returning all crimes. See \texttt{ukc\_crime\_category()} for details. See \texttt{ukc\_forces()} for details.

date  The year and month in "YYYY-MM" form. If NULL, latest available month will be returned. Also accepts dates in formats that can be coerced to \texttt{Date} class with \texttt{as.Date()}.

Value

A \texttt{tibble} with details of crimes without a specific location.

Examples

```r
## Not run:
no_location <- ukc\_crime\_no\_location(force = "city-of-london")
## End(Not run)
```

---

\textbf{ukc\_forces  \textit{England and Wales Police Forces}}

Description

Returns available police forces, details on a specific police force, or officers on that force.

Usage

- \texttt{ukc\_forces()}
- \texttt{ukc\_force\_details(force = NULL)}
- \texttt{ukc\_officers(force)}

Arguments

- \texttt{force}  The id of the police force, available from the id column returned by \texttt{ukc\_forces}. Not case sensitive.

Value

\texttt{ukc\_forces} returns a \texttt{tibble} with all police forces in England and Wales, \texttt{ukc\_force\_details} returns details on a given police force and \texttt{ukc\_officers} returns details on senior officers for a given police force.
ukc_neighbourhoods

Examples

```r
## Not run:
forces <- ukc_forces()
cops <- ukc_officers("cumbria")
details <- ukc_force_details("thames-valley")
## End(Not run)
```

ukc_last_update

Latest crime update

Description

Returns the latest month crime data was updated for. The date is in standard ISO format but the actual day is not relevant.

Usage

```r
ukc_last_update()
```

ukc_neighbourhoods

Policing Neighbourhoods

Description

All the neighbourhoods within a given police force area.

Usage

```r
ukc_neighbourhoods(force)
```

Arguments

- `force` A string containing the name of the police force to return neighbourhoods for. Must be specified, and is not case sensitive.

Value

A tibble with data for neighbourhoods within the area of the given police force.

See Also

- `ukc_neighbourhood_boundary()`
- `ukc_neighbourhood_specific()`
- `ukc_neighbourhood_location()`
ukc_neighbourhood_boundary

Examples

## Not run:
places <- ukc_neighbourhoods("dorset")

## End(Not run)

---

ukc_neighbourhood_boundary

*Specific Neighbourhood Boundary*

Description

Data on a specific neighbourhood boundary, using lat/lng pairs.

Usage

ukc_neighbourhood_boundary(force, neighbourhood_id)

Arguments

- **force**
  A string containing the name of the police force to return neighbourhoods for. Must be specified, and is not case sensitive.

- **neighbourhood_id**
  A string containing the ID of a given neighbourhood, returned from `ukc_neighbourhoods()`. If missing, returns all neighbourhoods for the specified police force, using `ukc_neighbourhoods()`.

Value

A tibble with the lat/lng boundaries for a specific neighbourhood.

See Also

`ukc_neighbourhoods()`

`ukc_neighbourhood_specific()`

`ukc_neighbourhood_location()`

Examples

## Not run:
borders <- ukc_neighbourhood_boundary("dorset", "55.CR3001")

## End(Not run)
**ukc_neighbourhood_location**  
*Neighbourhood Location*

**Description**

Find the neighbourhood policing team responsible for a given area, by a set of coordinates. Accepts only a single pair of latitude and longitude.

**Usage**

```r
ukc_neighbourhood_location(lat, lng)
```

**Arguments**

- `lat` Latitude
- `lng` Longitude

**Value**

The police force and neighbourhood code of the given coordinates.

**See Also**

- `ukc_neighbourhood_boundary()`
- `ukc_neighbourhoods()`

**Examples**

```r
## Not run:
find <- ukc_neighbourhood_location(lat = 51.500617, lng = -0.124629)

## End(Not run)
```

---

**ukc_neighbourhood_specific**  
*Specific Neighbourhood*

**Description**

Data on a specific neighbourhood within a given police force area.
ukc_neighbourhood_specific

Usage

ukc_neighbourhood_specific(force, neighbourhood_id)

ukc_neighbourhood_events(force, neighbourhood_id)

ukc_neighbourhood_priorities(force, neighbourhood_id)

ukc_neighbourhood_team(force, neighbourhood_id)

Arguments

force A string containing the name of the police force to return neighbourhoods for. Must be specified, and is not case sensitive.

neighbourhood_id A string containing the ID of a given neighbourhood, returned from ukc_neighbourhoods(). If missing, returns all neighbourhoods for the specified police force, using ukc_neighbourhoods().

Details

ukpolice contains the following functions for specific neighbourhoods:

- ukc_neighbourhood_specific – Basic information and description of a specific neighbourhood.
- ukc_neighbourhood_team – Brief biographies of police officers responsible for the specified neighbourhood.
- ukc_neighbourhood_events – Police public engagement events taking place in the specified neighbourhood.
- ukc_neighbourhood_priorities – Policing priorities for the specified neighbourhood.

Value

A list with data for a specific neighbourhood, or a tibble with all neighbourhood IDs if no neighbourhood is specified.

See Also

ukc_neighbourhood_boundary()
ukc_neighbourhoods()
ukc_neighbourhood_location()

Examples

## Not run:
# returns a specific neighbourhood
places4 <- ukc_neighbourhood_specific("dorset", "55.CR3001")

# returns all neighbourhoods as specific neighbourhood is unspecified.
places3 <- ukc_neighbourhood_specific("dorset")
ukc_specific_outcome

---

### Description

Returns the outcomes (case history) for a specified crime. The ID of a crime is a 64-character string, named `persistent_id` and returned by other methods.

### Usage

```r
ukc_specific_outcome(persistent_id)
```

### Arguments

- `persistent_id`  The 64 character string that is the unique ID of a particular crime.

### Value

Either a tibble with basic details of a crime if no outcome is available, or a list with basic details and a tibble containing the outcome if one is available.

### Examples

```r
## Not run:
no_location <- ukc_crime_no_location(force = "city-of-london")

crime_id <- no_location$persistent_id[[1]]

outcome <- ukc_specific_outcome(crime_id)
## End(Not run)
```
ukc_stop_search_force  

Stop and Searches by Police Force

Description

Returns details of stop and searches carried out by a particular police force. Note that the police force must be specified.

Usage

ukc_stop_search_force(force, date = NULL)

Arguments

force  
A string containing the name of the police force to return data for. Must be specified, and is not case sensitive. See ukc_forces() for details.

date  
The year and month in "YYYY-MM" form. Also accepts dates in formats that can be coerced to Date class with as.Date(). If NULL, the latest available month will be returned. Note that not all forces will necessarily have reported stop and search data for the latest month; check availability with the ukc_available() function.

Value

A tibble with details of stop and searches by a given police force.

See Also

ukc_stop_search_location()

ukc_stop_search_no_location()

Examples

## Not run:
ss_dorset <- ukc_stop_search_force(force = "dorset")

## End(Not run)
ukc_stop_search_loc  Stop and search

Description

Returns details on stops and searches at a given location. The stop and searches returned in the API, like the crimes, are only an approximation of where the actual stop and searches occurred, they are not the exact locations.

Usage

ukc_stop_search_loc(location, date = NULL, ...)
ukc_stop_search_coord(lat, lng, date = NULL, ...)
ukc_stop_search_poly(poly_df, date = NULL, ...)
ukc_stop_search_location(lat, lng, location, date = NULL)

Arguments

location  If specified, lat and lng are ignored. Location IDs are available through other methods including \texttt{ukc\_street\_crime()}.
date  The year and month in "YYYY-MM" form. If NULL, latest available month will be returned. Also accepts dates in formats that can be coerced to Date class with \texttt{as.Date()}.
...  further arguments passed to \texttt{httr::GET}.
l lat  Latitude. Accepts a single value or a vector of values to create a custom polygon.
lng  Longitude. Accepts a single value or a vector of values to create a custom polygon.
poly\_df  a dataframe containing the lat/lng pairs which define the boundary of the custom area, or a \texttt{sf} or \texttt{SpatialPointsDataFrame} object. The first and last coordinates need not be the same — they will be joined by a straight line once the request is made. If a dataframe, the lat/lng must be the first two columns, or named "lat" and "lng".

Details

Functions accept one of lat and lng pairs, location IDs or a dataframe containing lat/lng pairs defining the boundary of a custom area.

If specified, lat and lng must be the same length. If only one set of coordinates are given, all recorded stop and searches within a one mile radius are returned. If multiple pairs, all recorded stop and searches within a custom drawn polygon will be returned.
ukc_stop_search_no_location

Value

A tibble with details of stop and searches outcomes.

See Also

ukc_stop_search_force()
ukc_stop_search_no_location()

Examples

```r
## Not run:
ukc_stop_search2 <- ukc_stop_search_coord(
  lat = c(52.268, 53.194, 52.130),
  lng = c(0.543, 0.238, 0.478)
)

poly_df_4 <- data.frame(
  lat = c(52.268, 52.794, 52.130, 52.000),
  long = c(0.543, 0.238, 0.478, 0.400)
)

ukc_data_poly_4 <- ukc_stop_search_poly(poly_df = poly_df_4)
```

## End(Not run)

---

ukc_stop_search_no_location

*Stop and Searches without location*

Description

Returns details of stop and searches that cannot be mapped to a particular location. Note that
the police force must be specified. For all stop and searches carried out by a police force, use
ukc_stop_search_force().

Usage

`ukc_stop_search_no_location(force, date = NULL)`

Arguments

- `force` A string containing the name of the police force to return data for. Must be
  specified, and is not case sensitive. See `ukc_forces()` for details.
date The year and month in "YYYY-MM" form. Also accepts dates in formats that can be coerced to Date class with as.Date(). If NULL, the latest available month will be returned. Note that not all forces will necessarily have reported stop and search data for the latest month; check availability with the ukc_available() function.

Value
A tibble with details of stop and searches without a specific location.

See Also
ukc_stop_search_force()
ukc_stop_search_location()

Examples
## Not run:
ss_no_location <- ukc_stop_search_no_location(force = "city-of-london")
## End(Not run)
Value

A tibble with details of street crimes.

Examples

```r
## Not run:
crime <- ukc_street_crime(
  lat = 51.5, lng = -0.6,
  crime_category = "bicycle-theft"
)

crime_poly <- ukc_street_crime(
  lat = c(52.268, 52.794, 52.130),
  lng = c(0.543, 0.238, 0.478)
)
## End(Not run)
```

ukc_street_crime_outcome

*Street level crime outcomes*

Description

Returns details on crimes at a given location, if given the id of a specific location. If given latitude and longitude, finds the nearest pre-defined location and returns the crimes which occurred there.

Usage

```r
ukc_street_crime_outcome(lat, lng, location, date = NULL)
```

Arguments

- `lat` Latitude. Accepts a single value or a vector of values to create a custom polygon.
- `lng` Longitude. Accepts a single value or a vector of values to create a custom polygon.
- `location` If specified, `lat` and `lng` are ignored. Location IDs are available through other methods including `ukc_street_crime()`.
- `date` The year and month in "YYYY-MM" form. If NULL, latest available month will be returned.

Details

If specified, `lat` and `lng` must be the same length. `location` or both `lat` and `lng` must be specified.
Value

A tibble with details of street crime outcomes.

Examples

```r
## Not run:
street_crime_outcome1 <- ukc_street_crime_outcome(location = 883498)
street_crime_outcome2 <- ukc_street_crime_outcome(lat = 52, lng = 0)
## End(Not run)
```
Index

httr::GET, 3, 12

sf, 4, 12  
SpatialPointsDataFrame, 4, 12

ukc_available, 2  
ukc_available(), 11, 14

ukc_crime_category, 3  
ukc_crime_category(), 5, 14

ukc_crime_coord (ukc_crime_loc), 3

ukc_crime_location (ukc_crime_loc), 3

ukc_crime_no_location, 4

ukc_crime_poly (ukc_crime_loc), 3

ukc_force_details (ukc_forces), 5

ukc_forces(), 5, 11, 13

ukc_last_update, 6

ukc_neighbourhood_boundary, 7

ukc_neighbourhood_boundary(), 6, 8, 9

ukc_neighbourhood_events

(ukc_neighbourhood_specific), 8

ukc_neighbourhood_location, 8

ukc_neighbourhood_location(), 6, 7, 9

ukc_neighbourhood_priorities

(ukc_neighbourhood_specific), 8

ukc_neighbourhood_specific, 8

ukc_neighbourhood_specific(), 6, 7

ukc_neighbourhood_team

(ukc_neighbourhood_specific), 8

ukc_neighbourhoods, 6

ukc_neighbourhoods(), 7–9

ukc_officers (ukc_forces), 5

ukc_specific_outcome, 10

ukc_specific_outcome(), 2

ukc_stop_search_coord

(ukc_stop_search_loc), 12

ukc_stop_search_force, 11

ukc_stop_search_force(), 13, 14

ukc_stop_search_loc, 12

ukc_stop_search_location

(ukc_stop_search_loc), 12

ukc_stop_search_location(), 11, 14

ukc_stop_search_no_location, 13

ukc_stop_search_no_location(), 11, 13

ukc_stop_search_poly

(ukc_stop_search_loc), 12

ukc_street_crime, 14

ukc_street_crime(), 3, 12, 15

ukc_street_crime_outcome, 15

ukpolice-package, 2