

# Package ‘airportr’

October 9, 2019

**Type** Package

**Title** Convenience Tools for Working with Airport Data

**Version** 0.1.3

**Maintainer** Dmitry Shkolnik <shkolnikd@gmail.com>

**Description** Retrieves open source airport data and provides tools to look up information, translate names into codes and vice-verse, as well as some basic calculation functions for measuring distances. Data is licensed under the Open Database License.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Imports** dplyr

**Depends** R(>= 2.10.0)

**URL** <https://github.com/dshkol/airportr>

**BugReports** <https://github.com/dshkol/airportr/issues>

**RoxygenNote** 6.1.1

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Dmitry Shkolnik [cre, aut]

**Repository** CRAN

**Date/Publication** 2019-10-09 04:10:02 UTC

## R topics documented:

airportr	2
airports	2
airports_around	3
airports_near_airport	4
airport_detail	4

airport_distance . . . . .	5
airport_location . . . . .	5
airport_lookup . . . . .	6
city_airports . . . . .	7

<b>Index</b>	<b>8</b>
--------------	----------

---

airportr	airportr <i>package</i>
----------	-------------------------

---

### Description

Package to work with airport data

### Details

See the README on [GitHub](#)

---

airports	<i>Table of airport detail data</i>
----------	-------------------------------------

---

### Description

A dataset containing names, codes, locations, altitude, and timezones for airports

### Usage

```
airports
```

### Format

A data frame with 7698 rows and 14 variables:

**OpenFlights ID** OpenFlights database ID

**Name** Airport name, sometimes contains name of the city

**City** Name of city served by airport

**IATA** 3-letter IATA code

**ICAO** 4-letter ICAO code

**Country** Country name as in OpenFlights database. Note that country names may not be ISO 3166-1 standard.

**Country Code** ISO 3166-1 numeric country code

**Country Code (Alpha-2)** Name of city served by airport

**Country Code (Alpha-3)** Name of country where airport is located

**Latitude** Latitude in decimal degrees

**Longitude** Longitude in decimal degrees

**Altitude** Altitude in feet

**UTC** Hours offset from UTC

**DST** Daylight savings time. One of E (Europe), A (US/Canada), S (South America), O (Australia), Z (New Zealand), N (None) or U (Unknown)

**Timezone** Timezone in Olson format

**Type** Type of airport

**Source** Source of data. Airport data generally sourced from OurAirports

## Source

<https://openflights.org/data.html>

---

airports_around	<i>Lookup airports near specified coordinates</i>
-----------------	---

---

## Description

A function that returns details of all airports within a user-specified distance of an input coordinate location. Takes as input a longitude and latitude argument.

## Usage

```
airports_around(lat, lon, distance = 100)
```

## Arguments

lat	Latitude in decimal degrees
lon	Longitude in decimal degrees
distance	Distance boundary for nearest airport lookup in kilometres

## Value

A tibble of airports that fall within the specified range of specified location

## Examples

```
airports_around(-123, 49.2)

# Or with a user specified distance in kilometres
airports_around(-123, 49.2, distance = 200)
```

---

airports\_near\_airport *Lookup airports nearby other airports*

---

### Description

A function that returns details of airports within a user-specified distance of a given airport.

### Usage

```
airports_near_airport(input, distance = 100)
```

### Arguments

input	An airport name, IATA code, or ICAO code. Input type will be guessed unless explicitly defined
distance	Distance boundary for nearest airport lookup in kilometres

### Value

A tibble of airports that fall within the specified range of input airport

### Examples

```
airports_near_airport("YVR")

# Or with a user specified distance in kilometres
airports_near_airport("YVR", distance = 200)
```

---

airport\_detail *Lookup full airport details based of a standard airport input*

---

### Description

Return all airport details given an input IATA code, ICAO code, or airport name.

### Usage

```
airport_detail(input, input_type)
```

### Arguments

input	An airport name, IATA code, or ICAO code. Input type will be guessed unless explicitly defined
input_type	One of "name", "IATA", or "ICAO". Function will attempt to guess type if not supplied

**Value**

A 1x14 tibble with airport details

**Examples**

```
airport_detail("YVR")  
airport_detail("London Heathrow Airport")
```

---

airport_distance	<i>Calculate great circle distance between two airports</i>
------------------	---

---

**Description**

A function that calculates distances between pairs of airport codes. Distances are calculated using the Haversine formula which assumes a spherical earth. Distances are returned in kilometres.

**Usage**

```
airport_distance(airport1, airport2)
```

**Arguments**

airport1	Takes a three-letter IATA code corresponding to an airport
airport2	As above

**Value**

The great circle distance in kilometres between the two airports

**Examples**

```
airport_distance("YVR", "YYZ")
```

---

airport_location	<i>Lookup airport location coordinates given a standard airport input.</i>
------------------	--

---

**Description**

Returns airport location in longitude and latitude coordinates given an input IATA code, ICAO code, or airport name.

**Usage**

```
airport_location(input, input_type)
```

**Arguments**

input	An airport name, IATA code, or ICAO code. Input type will be guessed unless #' explicitly defined
input_type	One of "name", "IATA", or "ICAO". Function will attempt to guess type if not supplied

**Value**

List of longitude and latitude coordinates

**Examples**

```
airport_location("YVR","IATA")
#' airport_location("Vancouver International Airport","name")
```

---

airport_lookup	<i>Translate airport codes or names into other standard airport formats</i>
----------------	---

---

**Description**

Return city name, airport name, IATA code, or IACO code given an input IATA code, ICAO code, or airport name.

**Usage**

```
airport_lookup(input, input_type = "IATA", output_type = "name")
```

**Arguments**

input	An airport name, IATA code, or ICAO code. Input type will be guessed unless #' explicitly defined
input_type	One of "name", "IATA", or "ICAO". Function will attempt to guess type if not supplied
output_type	One of "name", "city", "IATA", or "ICAO". Defaults to "name" if otherwise not specified

**Value**

The appropriate city, airport name, IATA code, or ICAO code for that airport

**Examples**

```
airport_lookup("CYVR")
airport_lookup("YVR", output_type = "city")
airport_lookup("Vancouver International Airport", input_type="name",output_type = "IATA")
airport_lookup("YVR",input_type = "IATA", output_type = "city")

# Produces a list of similar named airports
airport_lookup("Vancoover","name","city")
```

---

city_airports	<i>Return all airports serving an input city.</i>
---------------	---

---

### Description

This function takes a city normal city name as an input argument and returns all airports associated with that city. Airports are typically associated with their local metropolitan area but some exceptions may be present in the data. If there are no matching results in the data for the city argument, a list of closely named alternatives will be suggested with a warning.

### Usage

```
city_airports(city, country)
```

### Arguments

city	A city name. If no exact match will attempt to prompt user with suggested alternatives
country	(Optional) A country name or ISO country code in either numeric, alpha-2, or alpha 3 format. Case insensitive.

### Value

A Nx17 tibble with airport details where n is the number of airports serving that city

### Examples

```
city_airports("Vancouver")
city_airports("London")
city_airports("London", "Canada")
city_airports("London", "CA")
city_airports("London", "CAN")
city_airports("London", "124")
```

# Index

## \*Topic **datasets**

- airports, [2](#)
  
- airport\_detail, [4](#)
- airport\_distance, [5](#)
- airport\_location, [5](#)
- airport\_lookup, [6](#)
- airportr, [2](#)
- airportr-package (airportr), [2](#)
- airports, [2](#)
- airports\_around, [3](#)
- airports\_near\_airport, [4](#)
  
- city\_airports, [7](#)