

Package ‘aopdata’

April 30, 2021

Title Data from the 'Access to Opportunities Project (AOP)'

Version 0.2.2

Date 2021-04-27

Description Download data from the 'Access to Opportunities Project (AOP)'. The 'aopdata' package brings annual estimates of access to employment, health and education services by transport mode, as well as data on the spatial distribution of population, schools and health-care facilities at a fine spatial resolution for all cities included in the study. More info on the 'AOP' website <<https://www.ipea.gov.br/acessoportunidades/en/>>.

URL <https://github.com/ipeaGIT/aopdata>

BugReports <https://github.com/ipeaGIT/aopdata/issues>

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 3.5.0)

Suggests covr, dplyr (>= 0.8-3), ggplot2 (>= 3.3.1), knitr, magrittr, mapview, rmarkdown (>= 2.6), scales, testthat, units

Imports curl, crul, data.table, httr (>= 1.4.1), sf (>= 0.9-3), utils

RoxygenNote 7.1.1

VignetteBuilder knitr

NeedsCompilation no

Author Rafael H. M. Pereira [aut, cre]
(<<https://orcid.org/0000-0003-2125-7465>>),
Daniel Herszenhut [aut] (<<https://orcid.org/0000-0001-8066-1105>>),
Marcus Saraiva [aut] (<<https://orcid.org/0000-0001-6218-2338>>),
Ipea - Institute for Applied Economic Research [cph, fnd]

Maintainer Rafael H. M. Pereira <rafa.pereira.br@gmail.com>

Repository CRAN

Date/Publication 2021-04-30 08:50:23 UTC

R topics documented:

aop_merge	2
aop_spatial_join	3
check_connection	3
download_data	4
download_metadata	4
load_data	5
read_access	5
read_grid	7
read_landuse	8
read_population	10
select_city_input	11
select_metadata	12
select_mode_input	13
select_year_input	13

Index	15
--------------	-----------

aop_merge	<i>Merge land use and access data</i>
-----------	---------------------------------------

Description

Merges landuse and access data

Usage

```
aop_merge(aop_landuse, aop_access)
```

Arguments

aop_landuse	A data.table of aop land use data
aop_access	A data.table of aop access data

Value

Returns a data.table with landuse and access data

See Also

Other support functions: [aop_spatial_join\(\)](#), [check_connection\(\)](#), [download_data\(\)](#), [load_data\(\)](#), [select_city_input\(\)](#), [select_metadata\(\)](#), [select_mode_input\(\)](#), [select_year_input\(\)](#)

aop_spatial_join	<i>Spatial join of AOP data</i>
------------------	---------------------------------

Description

Merges landuse or access data with H3 grid geometries

Usage

```
aop_spatial_join(aop_df, aop_sf)
```

Arguments

aop_df	A data.frame of aop data
aop_sf	A spatial sf of aop data

Value

Returns a data.frame sf with access/landuse data and grid geometries

See Also

Other support functions: [aop_merge\(\)](#), [check_connection\(\)](#), [download_data\(\)](#), [load_data\(\)](#), [select_city_input\(\)](#), [select_metadata\(\)](#), [select_mode_input\(\)](#), [select_year_input\(\)](#)

check_connection	<i>Check internet connection with Ipea server</i>
------------------	---

Description

Checks if there is internet connection to Ipea server to download aop data.

Usage

```
check_connection(  
  file_url = "https://www.ipea.gov.br/geobr/aopdata/metadata/metadata.csv"  
)
```

Arguments

file_url	A string with the file_url address of an aop dataset
----------	--

Value

Logic TRUE or FALSE'.

See Also

Other support functions: [aop_merge\(\)](#), [aop_spatial_join\(\)](#), [download_data\(\)](#), [load_data\(\)](#), [select_city_input\(\)](#), [select_metadata\(\)](#), [select_mode_input\(\)](#), [select_year_input\(\)](#)

download_data	<i>Download data to temporary directory.</i>
---------------	--

Description

Save requested data (either an sf or a data.frame) to a temporary directory.

Usage

```
download_data(file_url, progress_bar = showProgress)
```

Arguments

file_url	A string with the file_url address of a aop dataset
progress_bar	Logical. Defaults to (TRUE) display progress bar

Value

No visible output. The downloaded file (either an sf or a data.frame) is saved to a temporary directory.

See Also

Other support functions: [aop_merge\(\)](#), [aop_spatial_join\(\)](#), [check_connection\(\)](#), [load_data\(\)](#), [select_city_input\(\)](#), [select_metadata\(\)](#), [select_mode_input\(\)](#), [select_year_input\(\)](#)

download_metadata	<i>Support function to download metadata internally used in 'aopdata'</i>
-------------------	---

Description

Support function to download metadata internally used in 'aopdata'

Usage

```
download_metadata()
```

Value

A data.frame object with metadata and url of data sets

Examples

```
df <- download_metadata()
```

load_data	<i>Load data from tempdir to global environment</i>
-----------	---

Description

Reads data from tempdir to global environment.

Usage

```
load_data(file_url, temps = NULL)
```

Arguments

file_url	A string with the file_url address of a aop dataset
temps	The address of a data file stored in tempdir. Defaults to NULL

Value

Returns either an `sf` or a `data.frame`, depending of the data set that was downloaded

See Also

Other support functions: [aop_merge\(\)](#), [aop_spatial_join\(\)](#), [check_connection\(\)](#), [download_data\(\)](#), [select_city_input\(\)](#), [select_metadata\(\)](#), [select_mode_input\(\)](#), [select_year_input\(\)](#)

read_access	<i>Download accessibility estimates with population and land use data</i>
-------------	---

Description

Download annual estimates of access to employment, health and education services by transport mode and time of the day. See documentation 'Details' for the data dictionary. The data comes aggregated on a hexagonal grid based on the global H3 index at resolution 8, with a size of 357 meters (short diagonal) and an area of 0.74 km². More information about H3 at <https://h3geo.org/docs/core-library/restable/>.

Usage

```
read_access(
  city,
  mode = "walk",
  peak = TRUE,
  year = 2019,
  geometry = FALSE,
  showProgress = TRUE
)
```

Arguments

city	Character. A city name or three-letter abbreviation. If city="all", results for all cities are loaded.
mode	Character. A transport mode. Modes available include 'public_transport', 'bicycle', or 'walk' (the default).
peak	Logical. If TRUE (the default), returns accessibility estimates during peak time, between 6am and 8am. If FALSE, returns accessibility during off-peak, between 2pm and 4am. This argument only takes effect when mode = public_transport.
year	Numeric. A year number in YYYY format. Default set to 2019, the only year currently available.
geometry	Logical. If FALSE (the default), returns a regular data.table of aop data. If TRUE, returns a an sf data.frame with simple feature geometry of spatial hexagonal grid H3. See details in read_grid .
showProgress	Logical. Defaults to TRUE display progress bar

Value

A data.frame object

Data dictionary:

The name of the columns with accessibility estimates are the junction of three components: 1) Indicator 2) Type of opportunity 3) Time threshold (if applicable)

1) Indicator:

Indicator	Description	Note
CMA	Cumulative opportunity measure (active)	
TMI	Travel time to closest opportunity	Value = Inf when travel time is longer than 2h (public transport) or 1,5

2) Type of opportunity:

Indicator	Description	Note
TT	All jobs	
TQ	Total jobs with partial match between job education and income quintile	
TD	Total jobs with partial match between job education and income decile	

ST	All healthcare facilities
SB	Healthcare facilities - Low complexity
SM	Healthcare facilities - Medium complexity
SA	Healthcare facilities - High complexity
ET	All public schools
EI	Public schools - early childhood
EF	Public schools - elementary schools
EM	Public schools - high schools

3) Time threshold (only applicable to CMA estimates):

Time threshold	Description	Note - Only applicable to:
15	Opportunities accessible within 15 min.	Active transport modes
30	Opportunities accessible within 30 min.	All transport modes
45	Opportunities accessible within 45 min.	Active transport modes
60	Opportunities accessible within 60 min.	All transport modes
90	Opportunities accessible within 90 min.	Public transport
120	Opportunities accessible within 120 min.	Public transport

Examples

```
# Read accessibility estimates of a single city
df <- read_access(city = 'Fortaleza', mode = 'public_transport', year = 2019, showProgress = FALSE)
df <- read_access(city = 'for', mode = 'public_transport', year = 2019, showProgress = FALSE)

# Read accessibility estimates for all cities
all <- read_access(city = 'all', mode = 'walk', year = 2019, showProgress = FALSE)
```

read_grid

Download spatial hexagonal grid H3

Description

Results of the AOP project are spatially aggregated on a hexagonal grid based on the global H3 index at resolution 8, with a size of 357 meters (short diagonal) and an area of 0.74 km². More information about H3 at <https://h3geo.org/docs/core-library/restable/>. See documentation 'Details' for the data dictionary.

Usage

```
read_grid(city, showProgress = FALSE)
```

Arguments

city	Character. A city name or three-letter abbreviation. If city="all", results for all cities are loaded.
showProgress	Logical. Defaults to TRUE display progress bar

Value

An sf data.frame object

Data dictionary:

	Data type	column	Description
	geographic	abbrev_muni	Abbreviation of city name (3 letters)
	geographic	name_muni	City name
	geographic	code_muni	7-digit code of each city
	geographic	id_hex	Unique id of hexagonal cell

Examples

```
# Read spatial grid of a single city
nat <- read_grid(city = 'Natal', showProgress = FALSE)

# Read spatial grid of all cities in the project
# all <- read_grid(city = 'all', showProgress = FALSE)
```

read_landuse

Download land use and population data

Description

Download data on the spatial distribution of population, schools and healthcare facilities at a fine spatial resolution for cities included in the study. The data comes aggregated on a hexagonal grid based on the global H3 index at resolution 8, with a size of 357 meters (short diagonal) and an area of 0.74 km². More information about H3 at <https://h3geo.org/docs/core-library/restable/>.

See documentation 'Details' for the data dictionary.

Usage

```
read_landuse(city = "bel", year = 2019, geometry = FALSE, showProgress = TRUE)
```


Arguments

city	Character. A city name or three-letter abbreviation. If city="all", results for all cities are loaded.
year	Numeric. A year number in YYYY format. Default set to 2019, the only year currently available.
geometry	Logical. If FALSE (the default), returns a regular data.table of aop data. If TRUE, returns a an sf data.frame with simple feature geometry of spatial hexagonal grid H3. See details in read_grid .
showProgress	Logical. Defaults to TRUE display progress bar

Value

A data.frame object or an sf data.frame object

Data dictionary:

Data type	column	Description	Value
geographic	abbrev_muni	Abbreviation of city name (3 letters)	
geographic	name_muni	City name	
geographic	code_muni	7-digit code of each city	
geographic	id_hex	Unique id of hexagonal cell	
sociodemographic	P001	Total number of residents	
sociodemographic	P002	Number of white residents	
sociodemographic	P003	Number of black residents	
sociodemographic	P004	Number of indiginous residents	
sociodemographic	P005	Number of asian-descendents residents	
sociodemographic	R001	Average household income per capita	R\$ (Brazilian Reais), values in 2
sociodemographic	R002	Income quintile group	1 (poorest), 2, 3, 4, 5 (richest)
sociodemographic	R003	Income decile group	1 (poorest), 2, 3, 4, 5, 6, 7, 8, 9,
land use	T001	Total number of formal jobs	
land use	T002	Total number of formal jobs with primary education	
land use	T003	Number of formal jobs with secondary education	
land use	T004	Number of formal jobs with tertiary education	
land use	E001	Total number of public schools	
land use	E002	Number of public schools - early childhood	
land use	E003	Number of public schools - elementary schools	
land use	E004	Number of public schools - high schools	
land use	S001	Total number of healthcare facilities	
land use	S002	Number of healthcare facilities - low complexity	
land use	S003	Number of healthcare facilities - medium complexity	
land use	S004	Number of healthcare facilities - high complexity	

Examples

```
# a single city
bho <- read_landuse(city = 'Belo Horizonte', year = 2019, showProgress = FALSE)
bho <- read_landuse(city = 'bho', year = 2019, showProgress = FALSE)

# all cities
all <- read_landuse(city = 'all', year = 2019)
```

read_population

Download population and socioeconomic data

Description

Download population and socioeconomic data from the Brazilian Census aggregated on a hexagonal grid based on the global H3 index at resolution 8, with a size of 357 meters (short diagonal) and an area of 0.74 km². More information about H3 at <https://h3geo.org/docs/core-library/restable/>.

Usage

```
read_population(
  city = "bel",
  year = 2010,
  geometry = FALSE,
  showProgress = TRUE
)
```

Arguments

city	Character. A city name or three-letter abbreviation. If city="all", results for all cities are loaded.
year	Numeric. A year number in YYYY format. Default set to 2019, the only year currently available.
geometry	Logical. If FALSE (the default), returns a regular data.table of aop data. If TRUE, returns a an sf data.frame with simple feature geometry of spatial hexagonal grid H3. See details in read_grid .
showProgress	Logical. Defaults to TRUE display progress bar

Value

A data.frame object or an sf data.frame object

Data dictionary:

Data type	column	Description	Value
geographic	abbrev_muni	Abbreviation of city name (3 letters)	
geographic	name_muni	City name	
geographic	code_muni	7-digit code of each city	
geographic	id_hex	Unique id of hexagonal cell	
sociodemographic	P001	Total number of residents	
sociodemographic	P002	Number of white residents	
sociodemographic	P003	Number of black residents	
sociodemographic	P004	Number of indiginous residents	
sociodemographic	P005	Number of asian-descendents residents	
sociodemographic	R001	Average household income per capita	R\$ (Brazilian Reais), values in 2010
sociodemographic	R002	Income quintile group	1 (poorest), 2, 3, 4, 5 (richest)
sociodemographic	R003	Income decile group	1 (poorest), 2, 3, 4, 5, 6, 7, 8, 9, 10 (richest)

Examples

```
# a single city
bho <- read_population(city = 'Belo Horizonte', year = 2010, showProgress = FALSE)
bho <- read_population(city = 'bho', year = 2010, showProgress = FALSE)

# all cities
all <- read_population(city = 'all', year = 2010)
```

select_city_input *Select city input*

Description

Subsets the metadata table by 'city'.

Usage

```
select_city_input(temp_meta = temp_meta, city = NULL)
```

Arguments

temp_meta	A dataframe with the file_url addresses of aop datasets
city	city input (passed from read_ function)

Value

A data.frame object with metadata subsetted by 'city'

See Also

Other support functions: [aop_merge\(\)](#), [aop_spatial_join\(\)](#), [check_connection\(\)](#), [download_data\(\)](#), [load_data\(\)](#), [select_metadata\(\)](#), [select_mode_input\(\)](#), [select_year_input\(\)](#)

select_metadata	<i>Select metadata</i>
-----------------	------------------------

Description

Subsets the metadata table by data 'type', 'city', 'year' and 'mode'

Usage

```
select_metadata(t = NULL, c = NULL, y = NULL, m = NULL)
```

Arguments

t	Type of data: 'access' or 'grid' (passed from read_ function)
c	City (passed from read_ function)
y	Year of the dataset (passed from read_ function)
m	Transport mode (passed from read_ function)

Value

A data.frame object with metadata subsetted by data type, 'city', 'year' and 'mode'

See Also

Other support functions: [aop_merge\(\)](#), [aop_spatial_join\(\)](#), [check_connection\(\)](#), [download_data\(\)](#), [load_data\(\)](#), [select_city_input\(\)](#), [select_mode_input\(\)](#), [select_year_input\(\)](#)

Examples

```
df <- download_metadata()
```

select_mode_input	<i>Select mode input</i>
-------------------	--------------------------

Description

Subsets the metadata table by 'mode'.

Usage

```
select_mode_input(temp_meta = temp_meta, mode = NULL)
```

Arguments

temp_meta	A dataframe with the file_url addresses of aop datasets
mode	Transport mode (passed by read_ function)

Value

A data.frame object with metadata subsetted by 'mode'

See Also

Other support functions: [aop_merge\(\)](#), [aop_spatial_join\(\)](#), [check_connection\(\)](#), [download_data\(\)](#), [load_data\(\)](#), [select_city_input\(\)](#), [select_metadata\(\)](#), [select_year_input\(\)](#)

select_year_input	<i>Select year input</i>
-------------------	--------------------------

Description

Subsets the metadata table by 'year'.

Usage

```
select_year_input(temp_meta = temp_meta, year = NULL)
```

Arguments

temp_meta	A dataframe with the file_url addresses of aop datasets
year	Year of the dataset (passed from read_ function)

Value

A data.frame object with metadata subsetted by 'year'

See Also

Other support functions: [aop_merge\(\)](#), [aop_spatial_join\(\)](#), [check_connection\(\)](#), [download_data\(\)](#), [load_data\(\)](#), [select_city_input\(\)](#), [select_metadata\(\)](#), [select_mode_input\(\)](#)

Index

* **accessibility data functions**

read_access, 5

* **general support functions**

download_metadata, 4

* **land use data functions**

read_landuse, 8

* **population data functions**

read_population, 10

* **spatial data functions**

read_grid, 7

* **support functions**

aop_merge, 2

aop_spatial_join, 3

check_connection, 3

download_data, 4

load_data, 5

select_city_input, 11

select_metadata, 12

select_mode_input, 13

select_year_input, 13

aop_merge, 2, 3–5, 12–14

aop_spatial_join, 2, 3, 4, 5, 12–14

check_connection, 2, 3, 3, 4, 5, 12–14

download_data, 2–4, 4, 5, 12–14

download_metadata, 4

load_data, 2–4, 5, 12–14

read_access, 5

read_grid, 6, 7, 9, 10

read_landuse, 8

read_population, 10

select_city_input, 2–5, 11, 12–14

select_metadata, 2–5, 12, 12, 13, 14

select_mode_input, 2–5, 12, 13, 14

select_year_input, 2–5, 12, 13, 13