Package ‘rfars’

November 1, 2022

Type Package
Title Download and Analyze Fatal Crash Data
Version 0.2.0
Description Download raw data from the Fatality Analysis Reporting System (<https://cdan.dot.gov/query>) and prepare it for research.
License CC0
Encoding UTF-8
LazyData true
Imports data.table, downloader, dplyr, janitor, lubridate, magrittr, readr, rlang, stringr, tidyselect, timetk
RoxygenNote 7.2.1
Depends R (>= 2.10)
Suggests knitr, rmarkdown, leaflet, leaflet.extras, ggplot2, scales, stargazer, viridis
VignetteBuilder knitr
URL https://github.com/s87jackson/rfars
BugReports https://github.com/s87jackson/rfars/issues
NeedsCompilation no
Author Steve Jackson [aut, cre]
Maintainer Steve Jackson <s87jackson@gmail.com>
Repository CRAN
Date/Publication 2022-11-01 08:34:53 UTC

R topics documented:

alcohol ................................................................. 2
bicyclist .............................................................. 3
counts ................................................................. 3
alcohol

(Internal) Find crashes involving alcohol

Description

These internal functions take the FARS object created by use_fars and look for various cases, such as distracted or drowsy drivers.

Usage

alcohol(FARS)

Arguments

FARS

The FARS data object to be searched.
bicyclist  \hspace{1cm} (Internal) Find crashes involving bicyclists

Description

These internal functions take the FARS object created by use_fars and look for various cases, such as distracted or drowsy drivers.

Usage

bicyclist(FARS)

Arguments

FARS  The FARS data object to be searched.

counts  \hspace{1cm} Generate counts

Description

Use FARS data to generate commonly requested counts.

Usage

counts(
    FARS,
    what = "crashes",
    years = NULL,
    interval = "year",
    where = NULL,
    who = NULL,
    involved = NULL,
    filterOnly = FALSE
)

Arguments

FARS  The input FARS object.
what  What to count: crashes, fatalities, or people involved.
years  The years over which to count.
interval  The interval in which to count: months or years.
where  Where to count: can specify rural/urban and/or state (e.g., where = "rural Virginia", where = "rural", where = "North Carolina")
who

The type of person to count: driver, passenger, pedestrian, bicyclist, or motorcyclist.

involved

Factors involved with the crash. Can be any of: distracted driver, drowsy driver, police pursuit, motorcycle, pedalcyclist, bicyclist, pedestrian, pedbike, young driver, older driver, speeding, alcohol, drugs, hit and run, roadway departure, rollover, or large trucks.

filterOnly

Logical, whether to only filter data or reduce to counts.

Value

Either a filtered tibble (filterOnly=TRUE) or a tibble of counts (filterOnly=FALSE). If filterOnly=TRUE, the tibble that is returned is the ‘flat’ tibble from the input FARS object, filtered according to other parameters.

Examples

get_fars(years = 2020, states="Virginia") %>%
  counts(
    where = "rural"
  )

get_fars(years = 2020, states="FL") %>%
  counts(
    involved = "older driver"
  )

get_fars(years = 2020, states = "21") %>%
  counts(
    involved = c("young driver", "alcohol")
  )

distracted_driver (Internal) Find crashes involving distracted drivers

Description

These internal functions take the FARS object created by use_fars and look for various cases, such as distracted or drowsy drivers.

Usage

distracted_driver(FARS)

Arguments

FARS The FARS data object to be searched.
download_fars

(Internal) Download FARS data files

Description

Download files from NHTSA, unzip, and prepare them.

Usage

download_fars(years, dest_raw, dest_prepd)

Arguments

years
  Years to be downloaded, in yyyy (character or numeric formats)
dest_raw
  Directory to store raw CSV files
dest_prepd
  Directory to store prepared CSV file

Details

Raw files are downloaded from NHTSA.

Value

Nothing, called for side effects.

drowsy_driver

(Internal) Find crashes involving drowsy drivers

Description

These internal functions take the FARS object created by use_fars and look for various cases, such as distracted or drowsy drivers.

Usage

drowsy_driver(FARS)

Arguments

FARS
  The FARS data object to be searched.
**drugs**

*(Internal) Find crashes involving drugs*

**Description**

These internal functions take the FARS object created by `use_fars` and look for various cases, such as distracted or drowsy drivers.

**Usage**

`drugs(FARS)`

**Arguments**

FARS

The FARS data object to be searched.

---

**fars_data_changes**

*Changes in FARS Data Elements by Data File and Year*

**Description**

A dataset describing major changes to the FARS data system over time.

**Usage**

`fars_data_changes`

**Format**

A data frame with 46 rows and 480 columns.

**Details**

See Appendix F of the *2020 Analytical User's Manual* for more information.

**Source**

[https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/813254](https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/813254)
**fars_data_structure**  

**Description**  
A dataset describing the structure and level of each raw FARS data file.

**Usage**  
fars_data_structure

**Format**  
A data frame with 27 rows and 4 columns.  
- **tablename**: the cleaned name of the data file  
- **structure**: either one or multiple, indicating the number of rows per entity  
- **level**: the entity level (crash, vehicle, or person) or the data file  
- **year_created**: the first year that the data file was in use

**Source**  

**fars_varnames**  

**Description**  
A dataset that translates machine-readable variable names to friendly names

**Usage**  
fars_varnames

**Format**  
A data frame with 468 rows and 4 columns

**Details**  

- **table**: the cleaned name of the data file  
- **original**: the original variable name  
- **friendly**: human-readable (friendly) version of the variable name  
- **original_clean**: the cleaned name of the variable
get_fars

Source

https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/813254

geo_relations

Synonym table for various geographical scales

Description

A dataset providing different ways to refer to states and counties.

Usage

geo_relations

Format

A data frame with 3,142 rows and 6 variables:

- **fips_state** 2-digit FIPS code indicating a state
- **fips_county** 3-digit FIPS code indicating a county within a state
- **fips_tract** 6-digit FIPS code indicating a tract within a county
- **state_name_abbr** 2-character, capitalized state abbreviation
- **state_name_full** fully spelled and case-sensitive state name
- **county_name_abbr** abbreviated county name (usually minus the word 'County')
- **county_name_full** fully spelled and case-sensitive county name

Source


get_fars

Get FARS data

Description

Bring FARS data into the current environment, whether by downloading it anew or by using pre-existing files.

Usage

get_fars(years = 2015:2020, states = NULL, dir = NULL, proceed = FALSE)
get_fars

Arguments

- **years**: Years to be downloaded, in yyyy (character or numeric formats), currently limited to 2015-2020 (the default).
- **states**: (Optional) States to keep. Leave as NULL (the default) to keep all states. Can be specified as full state name (e.g. "Virginia"), abbreviation ("VA"), or FIPS code (51).
- **dir**: Directory in which to search for or save a 'FARS data' folder. If NULL (the default), files are downloaded and unzipped to temporary directories and prepared in memory.
- **proceed**: Logical, whether or not to proceed with downloading files without asking for user permission (defaults to FALSE, thus asking permission)

Details

This function downloads raw data from NHTSA. If no directory (dir) is specified, raw CSV files are downloaded into the tempdir(), where they are also prepared, combined, and then brought into the current environment. If you specify a directory (dir), the function will look there for a 'FARS data' folder. If not found, it will be created and populated with raw and prepared CSV files. If the directory is found, the function makes sure all requested years are present and asks permission to download any missing years.

The object returned is a list with class 'FARS'. It has five tibbles: flat, multi_acc, multi_veh, multi_per, events.

Flat files are wide-formatted and presented at the person level. All crashes involve at least one motor vehicle, each of which may contain one or multiple people. These are the three entities of crash data. The flat files therefore repeat some data elements across multiple rows. Please conduct your analysis with your entity in mind.

Some data elements can include multiple values for any data level (e.g., multiple weather conditions corresponding to the crash, or multiple crash factors related to vehicle or person). These elements have been collected in the yyyy_multi_[acc/veh/per].csv files in long format. These files contain crash, vehicle, and person identifiers, and two variables labelled name and value. These correspond to variable names from the raw data files and the corresponding values, respectively.

The events tibble provides a sequence of events for all vehicles involved in the crash. See Crash Sequences vignette for an example.

Consult the Analytical User’s Manual for more information.

Value

A FARS data object (a list with five tibbles: flat, multi_acc, multi_veh, multi_per, events)

Examples

```r
myFARS <- get_fars(years = 2019:2020, states = "51")
myFARS <- get_fars(years = 2020, states = "NC")
```
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
<th>Usage</th>
<th>Arguments</th>
</tr>
</thead>
<tbody>
<tr>
<td>large_trucks</td>
<td>(Internal) Find crashes involving large trucks</td>
<td><code>large_trucks(FARS)</code></td>
<td><code>FARS</code></td>
</tr>
<tr>
<td>motorcycle</td>
<td>(Internal) Find crashes involving motorcycles</td>
<td><code>motorcycle(FARS)</code></td>
<td><code>FARS</code></td>
</tr>
<tr>
<td>older_driver</td>
<td>(Internal) Find crashes involving older drivers</td>
<td><code>older_driver(FARS)</code></td>
<td><code>FARS</code></td>
</tr>
</tbody>
</table>

These internal functions take the FARS object created by `use_fars` and look for various cases, such as distracted or drowsy drivers.
pedalcyclist (Internal) Find crashes involving pedalcyclists

Description
These internal functions take the FARS object created by use_fars and look for various cases, such as distracted or drowsy drivers.

Usage
pedalcyclist(FARS)

Arguments
FARS The FARS data object to be searched.

pedbike (Internal) Find crashes involving pedstrians or bicyclists

Description
These internal functions take the FARS object created by use_fars and look for various cases, such as distracted or drowsy drivers.

Usage
pedbike(FARS)

Arguments
FARS The FARS data object to be searched.

pedestrian (Internal) Find crashes involving pedestrians

Description
These internal functions take the FARS object created by use_fars and look for various cases, such as distracted or drowsy drivers.

Usage
pedestrian(FARS)

Arguments
FARS The FARS data object to be searched.
police_pursuit  (Internal) Find crashes involving police pursuits

Description

These internal functions take the FARS object created by use_fars and look for various cases, such as distracted or drowsy drivers.

Usage

police_pursuit(FARS)

Arguments

FARS  The FARS data object to be searched.

prep_fars_2015  Handle yyyy data preparation

Description

Handle yyyy data preparation

Usage

prep_fars_2015(y, wd, rawfiles, prepared_dir)

Arguments

y  year, to be passed from prep_fars
wd  working directory, , to be passed from prep_fars
rawfiles  dataframe translating filenames into standard terms, to be passed from prep_fars
prepared_dir  the location where prepared files will be saved, to be passed from prep_fars

Value

Produces five files: yyyy_flat.csv, yyyy_multi_acc.csv, yyyy_multi_veh.csv, yyyy_multi_per.csv, and yyyy_events.csv
**prep_fars_2017**  
Handle yyyy data preparation

**Usage**

```r
prep_fars_2017(y, wd, rawfiles, prepared_dir)
```

**Arguments**

- `y`: year, to be passed from `prep_fars`
- `wd`: working directory, to be passed from `prep_fars`
- `rawfiles`: dataframe translating filenames into standard terms, to be passed from `prep_fars`
- `prepared_dir`: the location where prepared files will be saved, to be passed from `prep_fars`

**Value**

Produces five files: `yyyy_flat.csv`, `yyyy_multi_acc.csv`, `yyyy_multi_veh.csv`, `yyyy_multi_per.csv`, and `yyyy_events.csv`
**Description**
Handle yyyy data preparation

**Usage**
```
prep_fars_2019(y, wd, rawfiles, prepared_dir)
```

**Arguments**
- `y`: year, to be passed from `prep_fars`
- `wd`: working directory, to be passed from `prep_fars`
- `rawfiles`: dataframe translating filenames into standard terms, to be passed from `prep_fars`
- `prepared_dir`: the location where prepared files will be saved, to be passed from `prep_fars`

**Value**
Produces five files: `yyyy_flat.csv`, `yyyy_multi_acc.csv`, `yyyy_multi_veh.csv`, `yyyy_multi_per.csv`, and `yyyy_events.csv`

**Description**
Handle yyyy data preparation

**Usage**
```
prep_fars_2020(y, wd, rawfiles, prepared_dir)
```

**Arguments**
- `y`: year, to be passed from `prep_fars`
- `wd`: working directory, to be passed from `prep_fars`
- `rawfiles`: dataframe translating filenames into standard terms, to be passed from `prep_fars`
- `prepared_dir`: the location where prepared files will be saved, to be passed from `prep_fars`

**Value**
Produces five files: `yyyy_flat.csv`, `yyyy_multi_acc.csv`, `yyyy_multi_veh.csv`, `yyyy_multi_per.csv`, and `yyyy_events.csv`
### read_basic_csv

**Description**

(Internal) Takes care of basic CSV reading

**Usage**

```r
read_basic_csv(x, wd, rawfiles)
```

**Arguments**

- `x` The cleaned name of the data table (CSV).
- `wd` The working directory for these files
- `rawfiles` The data frame connecting raw filenames to cleaned ones.

### speeding

**Description**

(Internal) Find crashes involving speeding

**Usage**

```r
speeding(FARS)
```

**Arguments**

- `FARS` The FARS data object to be searched.
use_fars

(Internal) Use FARS data files

Description

Combine multiple years of prepared FARS data stored in CSV files and bring into the current environment.

Usage

use_fars(prepared_dir = "FARS data", years = NULL, states = NULL)

Arguments

prepared_dir Directory where prepared files are currently saved.
years (Optional) Years to keep.
states (Optional) States to keep.

descriptions

(Internal) Use decoded variables instead of encoded ones

Description

The raw data files include two versions of many data elements: an encoded one (using numbers to indicate characteristics such as injury severity, relation to the roadway, race, etc.) and another that has already been decoded. These variables are labelled according to the convention: x and xname, where the latter is the decoded version. This internal function goes through a given data frame, removing the decoded versions and renaming the encoded ones to remove the 'name' suffix.

Usage

usenames(df)

Arguments

df Data frame with both versions of some variables.

Value

A data frame with the encoded variables replaced with decoded versions.

See Also

prep_fars()
validate_states

Value

Returns an object of class 'FARS' which is a list of five tibbles: flat, multi_acc, multi_veh, multi_per, and events

validate_states  (Internal) Validate user-provided list of states

Description

(Internal) Validate user-provided list of states

Usage

validate_states(states)

Arguments

states  States specified in get_fars, prep_fars, or counts

validate_years

(Internal) Validate user-provided list of states

Description

(Internal) Validate user-provided list of states

Usage

validate_years(years)

Arguments

years  Years specified in download_fars, get_fars, prep_fars, or counts
| young_driver | (Internal) *Find crashes involving young drivers* |

**Description**

These internal functions take the FARS object created by `use_fars` and look for various cases, such as distracted or drowsy drivers.

**Usage**

`young_driver(FARS)`

**Arguments**

- **FARS**
  
  The FARS data object to be searched.
Index

* datasets
  - fars_data_changes, 6
  - fars_data_structure, 7
  - fars_varnames, 7
  - geo_relations, 8

alcohol, 2
bicyclist, 3
counts, 3
distracted_driver, 4
download_fars, 5
drowsy_driver, 5
drugs, 6

  - fars_data_changes, 6
  - fars_data_structure, 7
  - fars_varnames, 7

geo_relations, 8
get_fars, 8

large_trucks, 10
motorcycle, 10
older_driver, 10
pedalcyclist, 11
pedbike, 11
pedestrian, 11
police_pursuit, 12
prep_fars_2015, 12
prep_fars_2017, 13
prep_fars_2018, 13
prep_fars_2019, 14
prep_fars_2020, 14
read_basic_csv, 15

speeding, 15
use_fars, 16
usenames, 16
validate_states, 17
validate_years, 17
young_driver, 18