trafficAccidents\_vignette.R

cm0027

2021-08-25

This package contains functions to retrieve crash information data from the US National Highway Traffic Safety Administration's Faulty Analysis Reporting System, which is a nationwide census providing the American public yearly data regarding fatal injuries suffered in motor vehicle traffic crashes. There are two main functions available in the package:

fars\_summarize\_years function can retrieve total number of crashes per month per year fars\_map\_state function plots the location of crashes per state per year

Reads a file and loads it into a tibble

This is a simple function that loads the file indicated by filename into a ’tbl\_df’ class (the ’tibble’) that provides stricter checking and better formatting than the traditional data frame.

@param filename A character string giving the name of the file to be loaded

@return This function first checks if the file exists, if it doesn't it will return a message indicating that the file does not exist, if the file exists, it returns the file contents formated into a tibble.

@examples fars\_read('test.csv') fars\_read('test.txt')

@export @importFrom readr read\_csv

fars\_read <- function(filename) {  
 if(!file.exists(filename))  
 stop("file '", filename, "' does not exist")  
 data <- suppressMessages({  
 readr::read\_csv(filename, progress = FALSE)  
 })  
 dplyr::tbl\_df(data)  
}

Generates a file name based on the input year

This is a simple function that produces the file name for the year requested

@param year A character string giving the year of file name to be generated

@return This function generates the file name (character string) corresponding to the year of interest. Available years are: 2013, 2014 and 2015.

@examples make\_filename('2013') make\_filename('2014')

@export

make\_filename <- function(year) {  
 year <- as.integer(year)  
 sprintf("accident\_%d.csv.bz2", year)  
}

Retrives MONTH and year column values for selected years

For each year listed in the input, the function reads a file related to the year, checks if the year requested appears in the file column 'year', and if so, retrieves the month information for that year. The output is a list of tibbles with two columns, one for MONTH and second for year.

@param years A list of character strings giving the years of the files to be querried.

@return This function returns the MONTH and year column values for the years (rows) indicated in the input list. If the year requested is invalid, it will return a message indicateing that the year is invalid.

@examples fars\_read\_years(list('2013','2014')) fars\_read\_years(list('2013'))

@export @importFrom dplyr mutate select

fars\_read\_years <- function(years) {  
 lapply(years, function(year) {  
 file <- make\_filename(year)  
 tryCatch({  
 dat <- fars\_read(file)  
 dplyr::mutate(dat, year = year) %>%  
 dplyr::select(MONTH, year)  
 }, error = function(e) {  
 warning("invalid year: ", year)  
 return(NULL)  
 })  
 })  
}

Summarises the number of accidents per month, per year.

This function retrives MONTH and year column values for selected years and summarises the number of accidents per month, per year

@param years A list of character strings giving the years of the files to be querried

@return Retrives MONTH and year column values for selected years and summarises the number of accidents per month, per year, in a tibble with n+1 columns, were n is the number of years in the input list. The extra columns corresponds to the MONTH. Each cell corresponds to the number of accidents per month (rows), per year (columns).

@examples fars\_summarize\_years(list('2013','2014')) fars\_summarize\_years(list('2013','2014','2015'))

@export @importFrom dplyr bind\_rows group\_by summarize @importFrom tidyr spread

fars\_summarize\_years <- function(years) {  
dat\_list <- fars\_read\_years(years)  
dplyr::bind\_rows(dat\_list) %>%  
 dplyr::group\_by(year, MONTH) %>%  
 dplyr::summarize(n = n()) %>%  
 tidyr::spread(year, n)  
}

Plots accidents location per state, per year.

This function plots a map of the selected state and marks the location of each accident during the selected year on the map.

@param state.num An integer with values 1 to 56 corresponding to the 56 states of the united states @param year A character strings indicating the year during which the accidents to be plotted occured

@return A plot of the selected state with dots indicating the location of the accidents occured during the selected year. If there are no accidents to plot a message indicating that there are no accidents to report is printed.

@examples fars\_map\_state(state.num=1, year='2013') fars\_map\_state(state.num=10, year='2014')

@export @importFrom dplyr filter @importFrom maps map @importFrom graphics points

fars\_map\_state <- function(state.num, year) {  
 filename <- make\_filename(year)  
 data <- fars\_read(filename)  
 state.num <- as.integer(state.num)  
  
 if(!(state.num %in% unique(data$STATE)))  
 stop("invalid STATE number: ", state.num)  
 data.sub <- dplyr::filter(data, STATE == state.num)  
 if(nrow(data.sub) == 0L) {  
 message("no accidents to plot")  
 return(invisible(NULL))  
 }  
 is.na(data.sub$LONGITUD) <- data.sub$LONGITUD > 900  
 is.na(data.sub$LATITUDE) <- data.sub$LATITUDE > 90  
 with(data.sub, {  
 maps::map("state", ylim = range(LATITUDE, na.rm = TRUE),  
 xlim = range(LONGITUD, na.rm = TRUE))  
 graphics::points(LONGITUD, LATITUDE, pch = 46)  
 })  
}