FARS package

Chau Nam Ky Nguyen

2018-02-03

The functions in this pacakage will be using data from the US National Highway Traffic Safety Administration's Fatality Analysis Reporting System, which is a nationwide census providing the American public yearly data regarding fatal injuries suffered in motor vehicle traffic crashes. This package contains following functions:

- make_filename
- fars_read
- fars_read_years
- fars_summarize_years
- fars_map_state

Data Info

Data in this package ranges from 2013 to 2015. They are stored in three CSV files:

- accident_2013.csv.bz2
- accident_2014.csv.bz2
- accident_2015.csv.bz2

For reading data user can use two functions `make_filename` to create a full path to a raw file, and `fars_read` to read the CSV file into the system.

```
library(fars)

file = make_filename(2013)
data = fars_read(file)
dim(data)
```

```
## [1] 30202    50
```

The data contains 30202 observations and 50 attributes. Attentionally, `fars_read_years` won't return the raw data, instead return the list of dataset for each year containing only two attributes with 12 rows for each month in a year.

```
library(fars)
data = fars_read_years(2013:2014)
```

```
## # A tibble: 30,202 x 50
##    STATE ST_CASE VE_TOTAL VE_FORMS PVH_INVL  PEDS PERNOTMVIT PERMVIT
##    <int>   <int>    <int>    <int>    <int> <int>      <int>   <int>
##  1     1   10001        1        1        0     0          0       8
##  2     1   10002        2        2        0     0          0       2
##  3     1   10003        1        1        0     0          0       1
##  4     1   10004        1        1        0     0          0       3
##  5     1   10005        2        2        0     0          0       3
##  6     1   10006        2        2        0     0          0       3
##  7     1   10007        1        1        0     0          0       1
##  8     1   10008        2        2        0     0          0       2
##  9     1   10009        1        1        0     0          0       1
## 10     1   10010        2        2        0     0          0       4
## # ... with 30,192 more rows, and 42 more variables: PERSONS <int>,
## #   COUNTY <int>, CITY <int>, DAY <int>, MONTH <int>, YEAR <int>,
## #   DAY_WEEK <int>, HOUR <int>, MINUTE <int>, NHS <int>, ROAD_FNC <int>,
## #   ROUTE <int>, TWAY_ID <chr>, TWAY_ID2 <chr>, MILEPT <int>,
## #   LATITUDE <dbl>, LONGITUD <dbl>, SP_JUR <int>, HARM_EV <int>,
## #   MAN_COLL <int>, RELJCT1 <int>, RELJCT2 <int>, TYP_INT <int>,
## #   WRK_ZONE <int>, REL_ROAD <int>, LGT_COND <int>, WEATHER1 <int>,
## #   WEATHER2 <int>, WEATHER <int>, SCH_BUS <int>, RAIL <chr>,
## #   NOT_HOUR <int>, NOT_MIN <int>, ARR_HOUR <int>, ARR_MIN <int>,
## #   HOSP_HR <int>, HOSP_MN <int>, CF1 <int>, CF2 <int>, CF3 <int>,
## #   FATALS <int>, DRUNK_DR <int>
## # A tibble: 30,056 x 50
##    STATE ST_CASE VE_TOTAL VE_FORMS PVH_INVL  PEDS PERNOTMVIT PERMVIT
##    <int>   <int>    <int>    <int>    <int> <int>      <int>   <int>
##  1     1   10001        1        1        0     0          0       2
##  2     1   10002        1        1        0     0          0       1
##  3     1   10003        2        2        0     0          0       7
##  4     1   10004        3        3        0     0          0       5
##  5     1   10005        1        1        0     0          0       1
##  6     1   10006        1        1        0     0          0       1
##  7     1   10007        3        2        1     0          0       2
##  8     1   10008        2        2        0     0          0       4
##  9     1   10009        3        3        0     0          0       5
## 10     1   10010        2        2        0     0          0       6
## # ... with 30,046 more rows, and 42 more variables: PERSONS <int>,
## #   COUNTY <int>, CITY <int>, DAY <int>, MONTH <int>, YEAR <int>,
## #   DAY_WEEK <int>, HOUR <int>, MINUTE <int>, NHS <int>, ROAD_FNC <int>,
## #   ROUTE <int>, TWAY_ID <chr>, TWAY_ID2 <chr>, MILEPT <int>,
## #   LATITUDE <dbl>, LONGITUD <dbl>, SP_JUR <int>, HARM_EV <int>,
## #   MAN_COLL <int>, RELJCT1 <int>, RELJCT2 <int>, TYP_INT <int>,
## #   WRK_ZONE <int>, REL_ROAD <int>, LGT_COND <int>, WEATHER1 <int>,
## #   WEATHER2 <int>, WEATHER <int>, SCH_BUS <int>, RAIL <chr>,
## #   NOT_HOUR <int>, NOT_MIN <int>, ARR_HOUR <int>, ARR_MIN <int>,
## #   HOSP_HR <int>, HOSP_MN <int>, CF1 <int>, CF2 <int>, CF3 <int>,
## #   FATALS <int>, DRUNK_DR <int>
```

```r
data[[1]]
```

```
## # A tibble: 30,202 x 2
##    MONTH  year
##    <int> <int>
##  1     1  2013
##  2     1  2013
##  3     1  2013
##  4     1  2013
##  5     1  2013
##  6     1  2013
##  7     1  2013
##  8     1  2013
##  9     1  2013
## 10     1  2013
## # ... with 30,192 more rows
```

Summary accidents in years

To summerize accidents in years, user can use `fars_summarize_years(years)` with `years` that is either an integer vector or a string vector. `years` is valid for only a period ranging from 2013 to 2015.

```r
library(fars)
fars_summarize_years(2013:2015)
```

```
## # A tibble: 30,202 x 50
##    STATE ST_CASE VE_TOTAL VE_FORMS PVH_INVL  PEDS PERNOTMVIT PERMVIT
##    <int>   <int>    <int>    <int>    <int> <int>      <int>   <int>
##  1     1   10001        1        1        0     0          0       8
##  2     1   10002        2        2        0     0          0       2
##  3     1   10003        1        1        0     0          0       1
##  4     1   10004        1        1        0     0          0       3
##  5     1   10005        2        2        0     0          0       3
##  6     1   10006        2        2        0     0          0       3
##  7     1   10007        1        1        0     0          0       1
##  8     1   10008        2        2        0     0          0       2
##  9     1   10009        1        1        0     0          0       1
## 10     1   10010        2        2        0     0          0       4
## # ... with 30,192 more rows, and 42 more variables: PERSONS <int>,
## #   COUNTY <int>, CITY <int>, DAY <int>, MONTH <int>, YEAR <int>,
## #   DAY_WEEK <int>, HOUR <int>, MINUTE <int>, NHS <int>, ROAD_FNC <int>,
## #   ROUTE <int>, TWAY_ID <chr>, TWAY_ID2 <chr>, MILEPT <int>,
## #   LATITUDE <dbl>, LONGITUD <dbl>, SP_JUR <int>, HARM_EV <int>,
## #   MAN_COLL <int>, RELJCT1 <int>, RELJCT2 <int>, TYP_INT <int>,
## #   WRK_ZONE <int>, REL_ROAD <int>, LGT_COND <int>, WEATHER1 <int>,
## #   WEATHER2 <int>, WEATHER <int>, SCH_BUS <int>, RAIL <chr>,
## #   NOT_HOUR <int>, NOT_MIN <int>, ARR_HOUR <int>, ARR_MIN <int>,
## #   HOSP_HR <int>, HOSP_MN <int>, CF1 <int>, CF2 <int>, CF3 <int>,
## #   FATALS <int>, DRUNK_DR <int>
## # A tibble: 30,056 x 50
##    STATE ST_CASE VE_TOTAL VE_FORMS PVH_INVL  PEDS PERNOTMVIT PERMVIT
##    <int>   <int>    <int>    <int>    <int> <int>      <int>   <int>
##  1     1   10001        1        1        0     0          0       2
##  2     1   10002        1        1        0     0          0       1
##  3     1   10003        2        2        0     0          0       7
##  4     1   10004        3        3        0     0          0       5
##  5     1   10005        1        1        0     0          0       1
##  6     1   10006        1        1        0     0          0       1
##  7     1   10007        3        2        1     0          0       2
##  8     1   10008        2        2        0     0          0       4
##  9     1   10009        3        3        0     0          0       5
## 10     1   10010        2        2        0     0          0       6
## # ... with 30,046 more rows, and 42 more variables: PERSONS <int>,
## #   COUNTY <int>, CITY <int>, DAY <int>, MONTH <int>, YEAR <int>,
## #   DAY_WEEK <int>, HOUR <int>, MINUTE <int>, NHS <int>, ROAD_FNC <int>,
## #   ROUTE <int>, TWAY_ID <chr>, TWAY_ID2 <chr>, MILEPT <int>,
## #   LATITUDE <dbl>, LONGITUD <dbl>, SP_JUR <int>, HARM_EV <int>,
## #   MAN_COLL <int>, RELJCT1 <int>, RELJCT2 <int>, TYP_INT <int>,
## #   WRK_ZONE <int>, REL_ROAD <int>, LGT_COND <int>, WEATHER1 <int>,
## #   WEATHER2 <int>, WEATHER <int>, SCH_BUS <int>, RAIL <chr>,
## #   NOT_HOUR <int>, NOT_MIN <int>, ARR_HOUR <int>, ARR_MIN <int>,
## #   HOSP_HR <int>, HOSP_MN <int>, CF1 <int>, CF2 <int>, CF3 <int>,
## #   FATALS <int>, DRUNK_DR <int>
## # A tibble: 32,166 x 52
##    STATE ST_CASE VE_TOTAL VE_FORMS PVH_INVL  PEDS PERNOTMVIT PERMVIT
##    <int>   <int>    <int>    <int>    <int> <int>      <int>   <int>
##  1     1   10001        1        1        0     0          0       1
##  2     1   10002        1        1        0     0          0       1
##  3     1   10003        1        1        0     0          0       2
##  4     1   10004        1        1        0     0          0       1
##  5     1   10005        2        2        0     0          0       2
##  6     1   10006        1        1        0     0          0       2
##  7     1   10007        1        1        0     0          0       2
##  8     1   10008        1        1        0     1          1       1
##  9     1   10009        1        1        0     0          0       1
## 10     1   10010        2        2        0     0          0       2
## # ... with 32,156 more rows, and 44 more variables: PERSONS <int>,
## #   COUNTY <int>, CITY <int>, DAY <int>, MONTH <int>, YEAR <int>,
## #   DAY_WEEK <int>, HOUR <int>, MINUTE <int>, NHS <int>, RUR_URB <int>,
## #   FUNC_SYS <int>, RD_OWNER <int>, ROUTE <int>, TWAY_ID <chr>,
## #   TWAY_ID2 <chr>, MILEPT <int>, LATITUDE <dbl>, LONGITUD <dbl>,
## #   SP_JUR <int>, HARM_EV <int>, MAN_COLL <int>, RELJCT1 <int>,
## #   RELJCT2 <int>, TYP_INT <int>, WRK_ZONE <int>, REL_ROAD <int>,
## #   LGT_COND <int>, WEATHER1 <int>, WEATHER2 <int>, WEATHER <int>,
## #   SCH_BUS <int>, RAIL <chr>, NOT_HOUR <int>, NOT_MIN <int>,
## #   ARR_HOUR <int>, ARR_MIN <int>, HOSP_HR <int>, HOSP_MN <int>,
## #   CF1 <int>, CF2 <int>, CF3 <int>, FATALS <int>, DRUNK_DR <int>
```

```
## # A tibble: 12 x 4
##    MONTH `2013` `2014` `2015`
##  * <int>  <int>  <int>  <int>
##  1     1   2230   2168   2368
##  2     2   1952   1893   1968
##  3     3   2356   2245   2385
##  4     4   2300   2308   2430
##  5     5   2532   2596   2847
##  6     6   2692   2583   2765
##  7     7   2660   2696   2998
##  8     8   2899   2800   3016
##  9     9   2741   2618   2865
## 10    10   2768   2831   3019
## 11    11   2615   2714   2724
## 12    12   2457   2604   2781
```

Display the accident in a state of a particular year

To plot accidents of a state in a particular year, we will use `fars_map_state(state.num, year)`. `state.num` indicates the number for one state, while `year` indicates which year you want to plot. These arguments must be scala and can be either integer or string.

For an example, to display accident in the state 4 of the year 2013, we can invoke as the below:

```r
library(fars)
fars_map_state("4", "2013")
```