

# Package ‘fuzzyMM’

January 27, 2015

**Type** Package

**Title** Map Matching Using Fuzzy Logic

**Version** 1.0.1

**Date** 2014-07-15

**Author** Nikolai Gorte

**Description** Implements a fuzzy-logic-based map-matching algorithm used to match GPS trajectories to the OpenStreetMap digital road network.

**Maintainer** Nikolai Gorte <n.gorte@gmail.com>

**Depends** R (>= 2.15.0), osmar, frbs,

**Imports** methods, igraph, rgeos, rgdal,

**Suggests** rjson, maptools, stringr, RCurl,

**License** GPL (>= 2)

**Collate** var\_bounds.R fuzzy\_model.R FIS1.R FIS2.R FIS3.R IMP.R SMP1.R  
SMP2.R DRN.R fuzzyMM.R Data-track.R fuzzyMM-package.R

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2014-07-18 20:20:05

## R topics documented:

fuzzyMM-package	2
create_drn	2
FIS_IMP	3
FIS_SMP1	4
FIS_SMP2	6
get_fis	7
get_var_bounds	8
mm	8
set_var_bounds	10
traj	11
update_mf	12

<b>Index</b>	<b>13</b>
--------------	-----------

---

fuzzyMM-package

*Map Matching Using a Fuzzy Logic Based Algorithm.*

---

### Description

fuzzyMM is a package that implements a fuzzy logic based map matching algorithm. More information about each function can be found in its help documentation.

### Details

Map matching is the process of matching inaccurate GPS trajectories to a digital road network. The algorithm implemented in this package uses fuzzy logic to solve this problem.

A detailed description of the fuzzy logic map matching algorithm can be found in Quddus (2006).

### Author(s)

Nikolai Gorte <n.gorte@gmail.com>

### References

Quddus, M. A. 2006 (January). *High Integrity Map Matching Algorithms for Advanced Transport Telematics Applications*. Ph.D. thesis, Imperial College London, United Kingdom.

### See Also

[mm](#)

---

create\_drn

*Digital Road Network*

---

### Description

Create a Digital Road Network

### Usage

```
create_drn(bbox)
```

### Arguments

bbox            spatial bounding box from spatial data ([bbox](#))

### Details

This function downloads OSM road data for the area defined by the bounding box and creates a digital road network.

**Author(s)**

Nikolai Gorte <n.gorte@gmail.com>

**Examples**

```
## Not run:
data(traj)
roads <- create_drn(bbox(traj))

# Plot everything
plot(traj)
lines(slot(roads, "s1"))

# Shortest path
shortest.paths(slot(roads, "g"), V(slot(roads, "g"))[1], V(slot(roads, "g"))[23])

## End(Not run)
```

---

FIS\_IMP

*Fuzzy Inference System 1 (FIS1)*

---

**Description**

Fuzzy Inference System used in the Initial Map Matching Process (IMP).

**Format**

frbs object

**Details**

The IMP is the first and very important part of the map matching process since it is responsible for the identification of the initial link. Here you can see the input variables, fuzzy subsets and fuzzy rules used in FIS1. It is useful to know the variables and how they affect the rule outputs in case you plan to change the range of the fuzzy subsets.

Input variables to this FIS are:

- speed of the vehicle,  $v$  (m/s)
- heading error, HE (degrees)
- perpendicular distance to candidate link, PD (m)
- horizontal dilution of precision (HDOP)

The fuzzy subsets of the input variables are:

- high, low, zero
- small, large
- short, long

- good, bad

The corresponding fuzzy rules used in this FIS are:

- If (  $v$  is high) and (HE is small) then (L1 is average)
- If ( $v$  is high) and (HE is large) then (L1 is low)
- If (HDOP is good) and (PD is short) then (L1 is average)
- If (HDOP is good) and (PD is long) then (L1 is low)
- If ( HE is small) and (PD is short) then ( L1 is high)
- If ( HE is large) and ( PD is long) then (L1 is low)

### See Also

[get\\_var\\_bounds](#), [set\\_var\\_bounds](#)

### Examples

```
fis_imp <- get_fis("IMP")
fis_imp
# Plot membership functions
plotMF(fis_imp)
```

---

FIS\_SMP1

*Fuzzy Inference System 2 (FIS2)*

---

### Description

Fuzzy Inference System used in the Subsequent Map Matching Process along a link (SMP-1).

### Format

frbs object

### Details

SMP-1 checks if the vehicle has already crossed a junction or is about to cross one and matches subsequent positions to the link identified by the IMP if that is not the case. Here you can see the input variables, fuzzy subsets and fuzzy rules used in FIS2. It is usefull to know the variables and how they affect the rule outputs in case you plan to change the range of the fuzzy subsets.

Input variables to this FIS are:

- speed of the vehicle,  $v$  (m/s)
- horizontal dilution of precision (HDOP)
- $\Delta d(m)$ , the difference between the distance from the last matched position to the downstream junction and the distance travelled by the vehicle since the last position fix
- heading increment, HI (degrees)

- $\alpha$  and  $\beta$  (degrees), location of the current position fix, relative to the link, seen from the last matched position and from the downstream junction respectively

The fuzzy subsets of the input variables are:

- high, low, zero
- good, bad
- positiv, negativ
- small, large, 180
- below90, above90

The corresponding fuzzy rules used in this FIS are:

- If ( $\alpha$  is below 90 ) and ( $\beta$  is below 90 degrees) then ( L2 is high)
- If ( $\Delta d$  is positive) and ( $\alpha$  is above 90 degrees) then ( L2 is low)
- If ( $\Delta d$  is positive) and ( $\beta$  is above 90 degrees) then ( L2 is low)
- If (HI is small) and ( $\alpha$  is below 90 degrees) and ( $\beta$  is below 90 degrees) then (L2 is high)
- If (HI is small) and ( $\Delta d$  is positive) and ( $\alpha$  is above 90 degrees) then (L2 is low)
- If (HI is small) and ( $\Delta d$  is positive) and ( $\beta$  is above 90 degrees) then (L2 is low)
- If (HI is large) and ( $\alpha$  is below 90 degrees) and ( $\beta$  is below 90 degrees) then (L2 is low)
- If (HDOP is good) and ( $v$  is zero) then (L2 is high)
- If (HDOP is good) and ( $\Delta d$  is negative) then (L2 is average)
- If (HDOP is good) and ( $\Delta d$  is positive) then (L2 is low)
- If ( $v$  is high) and (HI is small) then (L2 is average)
- If (HDOP is good) and ( $v$  is high) and (HI is 180 degrees) then (L2 is high)

### See Also

[get\\_var\\_bounds](#), [set\\_var\\_bounds](#)

### Examples

```

fis_smp1 <- get_fis("SMP1")
fis_smp1
# Plot membership functions
plotMF(fis_smp1)

```

**Description**

Fuzzy Inference System used in the Subsequent Map Matching Process at a junction (SMP-2).

**Format**

frbs object

**Details**

SMP-2 identifies a new link among the candidate links if the vehicle has crossed a junction. Here you can see the input variables, fuzzy subsets and fuzzy rules used in FIS3. It is useful to know the variables and how they affect the rule outputs in case you plan to change the range of the fuzzy subsets.

Input variables to this FIS are:

- speed of the vehicle,  $v$  (m/s)
- heading error, HE (degrees)
- perpendicular distance to candidate link, PD (m)
- horizontal dilution of precision (HDOP)
- link connectivity
- distance error

The fuzzy subsets of the input variables are:

- high, low, zero
- small, large
- short, long
- good, bad
- indirect, direct
- small2, large2

The corresponding fuzzy rules used in this FIS are:

- If ( $v$  is high) and (HE is small) then (L1 is average)
- If ( $v$  is high) and (HE is large) then (L1 is low)
- If (HDOP is good) and (PD is short) then (L1 is average)
- If (HDOP is good) and (PD is long) then (L1 is low)
- If ( HE is small) and (PD is short) then ( L1 is high)
- If ( HE is large) and ( PD is long) then (L1 is low)

- If (The connectivity with the previous link is low) then (The L3 is low)
- If (The connectivity with the previous link is high) then (The L3 is high)
- If (The distance error is low) then (The L3 is high)
- If (The distance error is high) then (The L3 is low)

**See Also**

[get\\_var\\_bounds](#), [set\\_var\\_bounds](#)

**Examples**

```
fis_smp2 <- get_fis("SMP2")
fis_smp2
# Plot membership functions
plotMF(fis_smp2)
```

---

get\_fis

*Get Fuzzy Inference System*

---

**Description**

Get the Fuzzy Inference System for IMP, SMP1 or SMP2.

**Usage**

```
get_fis(name = c("IMP", "SMP1", "SMP2"))
```

**Arguments**

name                    Name of the process: IMP, SMP1 or SMP2

**Author(s)**

Nikolai Gorte <n.gorte@gmail.com>

**Examples**

```
fis_imp <- get_fis("IMP")
fis_imp
# Plot membership functions
plotMF(fis_imp)
```

get\_var\_bounds      *Fuzzy subset range*

---

### Description

This function returns a data frame containing the bounds of the fuzzy subsets of each fuzzy variable.

### Usage

```
get_var_bounds()
```

### Details

Each fuzzy variable has fuzzy subsets. For example, in “speed is zero”, speed is the fuzzy variable and zero is the fuzzy subset.

The bounds of the fuzzy subset are used to create the membership functions, that are used to fuzzify the inputs of the fuzzy inference system. These bounds represent the x values at which the sigmoidal membership functions reach ~0 or ~1 respectively(e.g. “speed is high” ranges from 3 to 6 m/s).

### Author(s)

Nikolai Gorte <n.gorte@gmail.com>

### See Also

[FIS\\_IMP](#), [FIS\\_SMP1](#), [FIS\\_SMP2](#)

### Examples

```
get_var_bounds()
```

---

mm      *Map Matching*

---

### Description

Function that matches GPS trajectories to the OSM digital road network using a fuzzy logic map matching algorithm.

### Usage

```
mm(traj, plot = FALSE)
```



## Arguments

traj	<a href="#">SpatialPointsDataFrame-class</a> containing the GPS trajectories. See Details for additional info.
plot	boolean. Matched trajectory will be plotted if true.

## Details

**mm** is the main function of this package. The input to the function is a [SpatialPointsDataFrame-class](#) containing the GPS trajectory that needs to be matched to a digital road network.

To successfully apply the map matching algorithm the data part of the trajectories must include data for HDOP, Speed and Bearing, with the columns named "GPS.HDOP", "GPS.Speed" and "GPS.Bearing" respectively. Values for GPS.Speed must be given in km/h. Missing values in the data will be replaced with zeros and can lead to incorrect matchings.

The map data is obtained from OpenStreetMap.

The algorithm consists of three major parts responsible for the identification of the links the vehicle is travelling on.

- Initial MapMatching Process (IMP)
- Subsequent MapMatching Process along a link (SMP-1)
- Subsequent MapMatching Process at a junction (SMP-2)

Each of these processes uses a Fuzzy Inference System (FIS) for the link identification. Input variables, fuzzy subsets and fuzzy rules for each FIS can be seen in [FIS\\_IMP](#), [FIS\\_SMP1](#) and [FIS\\_SMP2](#) or by getting the corresponding [frbs](#) object using [get\\_fis](#).

A detailed description of the fuzzy logic map matching algorithm and the FIS can be found in Quddus (2006).

## Author(s)

Nikolai Gorte <n.gorte@gmail.com>

## References

Quddus, M. A. 2006 (January). *High Integrity Map Matching Algorithms for Advanced Transport Telematics Applications*. Ph.D. thesis, Imperial College London, United Kingdom.

## See Also

[FIS\\_IMP](#), [FIS\\_SMP1](#), [FIS\\_SMP2](#), [get\\_fis](#), [frbs](#)

## Examples

```
## Not run:  
data(traj)  
matched_traj <- mm(traj, plot = TRUE)  
  
## End(Not run)
```

---

set_var_bounds	<i>Set bounds of fuzzy subsets</i>
----------------	------------------------------------

---

### Description

This function allows to set the bounds for the fuzzy subsets to adjust the membership functions to your needs. These bounds represent the x values at which the sigmoidal membership functions reach ~0 or ~1 respectively(e.g. “speed is high” ranges from 3 to 6 m/s).

### Usage

```
set_var_bounds(name = c("speed_high", "speed_low", "speed_zero", "HE_small",  
  "HE_large", "PD_short", "PD_long", "HDOP_good", "HDOP_bad",  
  "alpha_low, alpha_high", "beta_low", "beta_high", "delta_dist_neg",  
  "delta_dist_pos", "HI_small", "HI_large", "HI_180", "connectivity_direct",  
  "connectivity_indirect", "dist_err_small", "dist_err_large"),  
  bounds = "numeric", default = FALSE)
```

### Arguments

name	name of the variable which bounds should be changed.
bounds	numeric vector containing the lower and upper bound.
default	logical, restores the default bounds if true.

### Author(s)

Nikolai Gorte <n.gorte@gmail.com>

### See Also

[FIS\\_IMP](#), [FIS\\_SMP1](#), [FIS\\_SMP2](#), [update\\_mf](#)

### Examples

```
set_var_bounds("speed_high", c(4, 7))  
get_var_bounds()  
update_mf()
```

---

traj	<i>enviroCar GPS trajectory</i>
------	---------------------------------

---

**Description**

A dataset containing GPS positions of a vehicle and data of several measured phenomenons.

**Format**

[SpatialPointsDataFrame-class](#) with 82 observations of 19 variables.

**Details**

- id.
- time.
- MAF: Mass Air Flow, l/s.
- Engine.Load, %.
- GPS.PDOP: Positional Dilution of Precision, precision.
- Intake.Temperature, deg. C.
- Calculated.MAF: Calculated Mass Air Flow, g/s.
- Throttle.Position, %.
- O2.Lambda.Voltage.ER, ratio.
- Rpm, u/min.
- GPS.Bearing, deg.
- GPS.Speed, km/h.
- O2.Lambda.Voltage, V.
- Intake.Pressure, kPa.
- GPS.Altitude, m.
- GPS.HDOP: Horizontal Dilution of Precision, precision.
- GPS.VDOP: Vertical Dilution of Precision, precision.
- Speed, km/h.
- GPS.Accuracy, %.

**Source**

enviroCar: <https://envirocar.org/api/stable/tracks/52f3836be4b0d8e8c27ed6f0>

---

`update_mf`*Update Membership Functions*

---

**Description**

This function updates the membership functions after the bounds of the fuzzy subsets are changed and reinitializes the fuzzy inference systems.

**Usage**`update_mf()`**Author(s)**

Nikolai Gorte <n.gorte@gmail.com>

**See Also**

[FIS\\_IMP](#), [FIS\\_SMP1](#), [FIS\\_SMP2](#)

**Examples**`update_mf()`

# Index

## \*Topic **datasets**

FIS\_IMP, [3](#)

FIS\_SMP1, [4](#)

FIS\_SMP2, [6](#)

traj, [11](#)

bbox, [2](#)

create\_drn, [2](#)

FIS\_IMP, [3](#), [8–10](#), [12](#)

FIS\_SMP1, [4](#), [8–10](#), [12](#)

FIS\_SMP2, [6](#), [8–10](#), [12](#)

frbs, [9](#)

fuzzyMM-package, [2](#)

get\_fis, [7](#), [9](#)

get\_var\_bounds, [4](#), [5](#), [7](#), [8](#)

mm, [2](#), [8](#)

set\_var\_bounds, [4](#), [5](#), [7](#), [10](#)

SpatialPointsDataFrame-class, [9](#), [11](#)

traj, [11](#)

update\_mf, [10](#), [12](#)