

Manual for Package: gis

Revision 4M

Karl Kästner

November 3, 2022

Contents

1	gis	1
1.1	GPX	1
1.2	batavia_zero	1
2	centreline/@Centreline	1
2.1	Centreline	1
2.2	channel_planimetry	1
2.3	clip	1
2.4	connect_graph	2
2.5	curvature	2
2.6	cut	2
2.7	determine_width	2
2.8	distance	2
2.9	export_cross_section	2
2.10	export_node	2
2.11	export_shp	2
2.12	find_nearest_segment	2
2.13	from_polygon	2
2.14	from_shp	3
2.15	get	3
2.16	init	3
2.17	init_connect	3
2.18	init_node_D	3
2.19	link_centreline	3
2.20	plot	3
2.21	plot_connection	3
2.22	prune	3
2.23	prune_leaves	3
2.24	prune_manually	4

2.25	reachable	4
2.26	remove_duplicate_points	4
2.27	resample	4
2.28	routing	4
2.29	routing2	4
2.30	shp_resample_simple	4
2.31	snmesh	4
2.32	squeeze	4
2.33	trim_ends	4
2.34	weighed_connection_matrix	5
2.35	xy2sn	5
3	centreline/@Segment	5
3.1	Segment	5
3.2	build_inverse_index	5
3.3	connectivity_matrix	5
3.4	init_seg_id	5
4	centreline	5
4.1	sn2xy_quadratic	5
4.2	thalweg	5
4.3	xy2sn_quadratic	5
5	gis	6
5.1	gpx_export_csv	6
5.2	hgt_plot	6
5.3	hgt_read	6
5.4	hgt_read_all	6
5.5	hgt_resample	6
5.6	nmeatime	6
5.7	read_xyz	6
6	shapefile/@Shp	6
6.1	Shp	6
6.2	area	7
6.3	buffer	7
6.4	centroid	7
6.5	clip	7
6.6	clip_rect	7
6.7	close_polygon	7
6.8	concat	7
6.9	connect_network	7
6.10	contour	8
6.11	copy_attribute	8

6.12	cp	8
6.13	create	8
6.14	curvature	8
6.15	cut	8
6.16	diameter	8
6.17	edges	8
6.18	export_geo	9
6.19	export_gpx	9
6.20	export_gpx_track	9
6.21	export_ldb	9
6.22	export_poly	9
6.23	export_sdf	9
6.24	export_spline	9
6.25	extract_coastline	9
6.26	first_point	10
6.27	flat	10
6.28	generate_four_colour_index	10
6.29	generate_rectangle	10
6.30	import_geo	10
6.31	import_poly	10
6.32	inpolygon	10
6.33	join_lines	10
6.34	last_point	11
6.35	latlon2utm	11
6.36	length	11
6.37	length2	11
6.38	line2point	11
6.39	link_lines	11
6.40	make_clockwise	11
6.41	merge	11
6.42	merge2	12
6.43	padd_nan	12
6.44	plot	12
6.45	points	12
6.46	polygon_boundary	12
6.47	read	12
6.48	readZ	12
6.49	remove_duplicate_points	12
6.50	remove_leaves	13
6.51	remove_nan	13
6.52	remove_polygon_closure	13
6.53	remove_short_elements	13
6.54	renumber	13
6.55	resample	13

6.56	resample_2	13
6.57	resample_min	13
6.58	resample_quick	14
6.59	scale	14
6.60	segment	14
6.61	select_for_refinement	14
6.62	set_geometry	14
6.63	set_resolution	14
6.64	singlepart_to_multipart	14
6.65	skip	14
6.66	smooth	15
6.67	split_jump	15
6.68	split_line	15
6.69	split_nan	15
6.70	swap_hemisphere	15
6.71	translate	15
6.72	write	15
7	shapefile	15
7.1	astar_multi	15
7.2	astar_recursive	16
7.3	edge_chain	16
7.4	edge_from_bnd	16
7.5	preload_shp	16
7.6	read_gpx	16
7.7	shapewrite_	16
7.8	shapewrite_man	18
7.9	shp2geo	18
7.10	shp2kml	19
7.11	shp_plot_attribute	19
7.12	split_section	19
7.13	write_polygon	19
8	gis	19
8.1	shp2csv	19
8.2	write_xyz	19

1 gis

1.1 GPX

1.2 `batavia_zero`

2 `centreline/@Centreline`

2.1 `Centreline`

2.2 `channel_planimetry`

2.3 `clip`

2.4 `connect_graph`

2.5 `curvature`

2.6 `cut`

2.7 `determine_width`

2.8 `distance`

2.9 `export_cross_section`

2.10 `export_node`

2.11 `export_shp`

2.12 `find_nearest_segment`

2.13 `from_polygon`

2.14 `from_shp`

2.15 `get`

2.16 `init`

`obj.seg_S(id(end)) = NaN;`

2.17 `init_connect`

2.18 `init_node_D`

2.19 `link_centreline`

2.20 `plot`

2.21 `plot_connection`

2.22 `prune`

2.23 `prune_leaves`

2.24 `prune_manually`

2.25 `reachable`

2.26 `remove_duplicate_points`

2.27 `resample`

2.28 `routing`

2.29 `routing2`

2.30 `shp_resample_simple`

2.31 `snmesh`

2.32 `squeeze`

2.33 `trim_ends`

2.34 `weighed_connection_matrix`

2.35 `xy2sn`

3 `centreline/@Segment`

3.1 `Segment`

3.2 `build_inverse_index`

3.3 connectivity_matrix

3.4 init_seg_id

4 centreline

4.1 sn2xy_quadratic

4.2 thalweg

4.3 xy2sn_quadratic

5 gis

5.1 gpx_export_csv

5.2 hgt_plot

5.3 hgt_read

```
% [ floor(median(z(kk))) meannan(z(kk)) min(z(kk)) max(z(kk)) ]
```

5.4 hgt_read_all

5.5 hgt_resample

5.6 nmeatime

5.7 read_xyz

6 shapefile/@Shp

6.1 Shp

6.2 area

area of polygon shapes

6.3 buffer

buffer or shrink a polygon by a fixed distance

6.4 centroid

6.5 clip

crop input shape file to specified polygon

6.6 clip_rect

rectrangular crop of the shapefile

6.7 close_polygon

close polygon, i.e. make the first point identical to the last

6.8 concat

concatenate two shapefiles

6.9 connect_network

```
TODO make unique
attach segments to
XY = [cvec(shp.X),shp.;
knnsearch for nearest n neighbours
for each segment
```

6.10 contour

6.11 copy_attribute

copy attributes from one shapefile to the other

6.12 cp

copy a shapefile on disk

6.13 create

create a new shapefile with given geometry

6.14 curvature

curvature of line segments

6.15 cut

6.16 diameter

determine diameter of polygon of every element

6.17 edges

edges of polygons line loops loops

6.18 export_geo

export geometry file understood by SLIM

6.19 export_gpx

export data into a gpx file

6.20 export_gpx_track

export a data into a gpx track file

6.21 export_ldb

export Delft3D-4 land-boundary

6.22 export_poly

export poly-file understood by SLIM

6.23 export_sdf

6.24 export_spline

export splines (for D3D?)

6.25 extract_coastline

6.26 first_point

extract first point of all shapefile features

6.27 flat

6.28 generate_four_colour_index

unique colour-indices fpr poligons

6.29 generate_rectangle

generate rectangular polygon

6.30 import_geo

6.31 import_poly

import poly file

6.32 inpolygon

test if point is in any of the polygons

6.33 join_lines

join line segments

6.34 last_point

return last point of features

6.35 latlon2utm

convert latitude and longitude to utm

6.36 length

number of points of each feature

6.37 length2

length of line segments

6.38 line2point

convert lines to points

6.39 link_lines

link lines with same endpoints

6.40 make_clockwise

make polygons clockwise

6.41 merge

6.42 merge2

6.43 padd_nan

padd NaN at end of features

6.44 plot

display the shapefile

6.45 points

returns points of the features

6.46 polygon_boundary

6.47 read

read shapefile from file

6.48 readZ

read shapefile with z-data from file
this is a workaround, as matlab cannot read files with z-data

6.49 remove_duplicate_points

remove duplicate points from features

6.50 remove_leaves

6.51 remove_nan

remove NaN points from features

6.52 remove_polygon_closure

remove last points of polygon if they are identical to the first

6.53 remove_short_elements

remove features with few points

6.54 renumber

generate a new index

6.55 resample

resample coordinates

6.56 resample_2

resample coordinates

6.57 resample_min

resample coordinates

6.58 resample_quick

resample coordinates

6.59 scale

6.60 segment

separate disjoint sections of polygons and lines

6.61 select_for_refinement

select elements for refinement

6.62 set_geometry

set feature geometry

6.63 `set_resolution`

set resolution for mesh generation

6.64 `singlepart_to_multipart`

concatenate line segments (parts) of shp data files into one
same as single part to multipart in qgis
returns also indices into the original file

6.65 `skip`

quick resampling of features by leaving out points

6.66 `smooth`

smooth the features

6.67 `split_jump`

split features where distance between points exceeds a threshold

6.68 `split_line`

split line features into single segments

6.69 `split_nan`

splits shp line and polygons at NaN into two different groups

6.70 `swap_hemisphere`

swap northern and southern hemisphere for UTM coordinates

6.71 translate

translate coordinates

6.72 write

write the shapefile to disk

7 shapefile

7.1 astar_multi

7.2 astar_recursive

astar path finding algorithm

7.3 edge_chain

7.4 edge_from_bnd

7.5 preload_shp

7.6 read_gpx

7.7 shapewrite__

Copyright (C) 2014,2015 Philip Nienhuis

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

-- texinfo --

@deftypefn {Function File} {@var{status} =} shapewrite (@var{shpstr}, @var{fname})

Write contents of map- or geostruct to a GIS shape file.

@var{shpstr} must be a valid mapstruct or geostruct, a struct array with an

entry for each shape feature, with fields Geometry, BoundingBox, and X and Y

(mapstruct) or Lat and Lon (geostruct). For geostructs, Lat and Lon field

data will be written as X and Y data. Field Geometry can have data values

of 'Point', 'MultiPoint', 'Line', or 'Polygon', all case-insensitive. For

each shape feature, field BoundingBox should contain the minimum and maximum

(X,Y) coordinates in a 2x2 array [minX, minY; maxX, maxY]. The X and Y

fields should contain X (or Latitude) and Y (or Longitude) coordinates for

each point or vertex as row vectors; for polylines and polygons vertices of

each subfeature (if present) should be separated by NaN entries.

@var{fname} should be a valid shape file name, optionally with a '.shp' suffix.

```

shapewrite produces 2 or 3 files, i.e. a .shp file (the actual
    shape file),
a .shx file (index file), and if @var{shpstr} contained additional
    fields,
a .dbf file (dBase type 3) with the contents of those additional
    fields.

@var{status} is 1 if the shape file set was written successfully, 0
otherwise.

@seealso{shaperead, shapeinfo}
@end deftypefn
Author: Philip Nienhuis <prnienhuis@users.sf.net>
Created: 2014-12-30
Input validation
Assess shape variable type (oct or ml/geo ml/map)
Yep. Find out what type
Assume it is an Octave-style struct read by shaperead
Assume it is a Matlab-style mapstruct
Assume it is a Matlab-style geostruct
Not a supported struct type
Check file name
Later on bname.shx and bname.dbf will be read
Prepare a few things
Change Lat/Lon fields into X/Y
Only now (after input checks) open .shp and .shx files & rewind
    just to be sure
Write headers in .shp & .shx (identical). First magic number 9994 +
    5 zeros
In between here = filelength in 16-bit words (single). For .shx it's
    known
Next, shp file version
Shape feature type
Bounding box. Can be run later for ML type shape structs. Fill with
    zeros
Skip to start of first record position
Write shape features one by one
Write record start pos to .shx file
Write record contents
Point
Record index number
Record length (fixed)
Shape type
Simply write XY coordinates
MultiPoint
Record index number
Record length
Shape type
Bounding box

```

Nr of points
Polyline/-gon
Record index number
Prepare multipart polygons
Augment idx for later on, & this trick eliminates trailing NaN rows
Record length
Shape type
Bounding box
Number of parts, number of points, part pointers
Write file length into .shp header
Close files
Check for dbfwrite function
Write rest of attributes
Attributes + shp data in mapstruct
Attributes + shp data in geostruct

7.8 shapewrite_man

7.9 shp2geo

7.10 shp2kml

7.11 shp_plot_attribute

7.12 split_section

7.13 write_polygon

8 `gis`

8.1 `shp2csv`

8.2 `write_xyz`