

# Manual for Package: gis

## Revision 1

Karl Kästner

March 28, 2020

## Contents

<b>1</b>	<b>gis</b>	<b>1</b>
1.1	GPX . . . . .	1
1.2	batavia_zero . . . . .	1
<b>2</b>	<b>centreline/@Centreline</b>	<b>1</b>
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
<b>3</b>	<b>centreline/@Segment</b>	<b>5</b>
3.1	Segment . . . . .	5
3.2	build_inverse_index . . . . .	5
3.3	connectivity_matrix . . . . .	5
3.4	init_seg_id . . . . .	5
<b>4</b>	<b>centreline</b>	<b>5</b>
4.1	sn2xy_quadratic . . . . .	5
4.2	thalweg . . . . .	5
4.3	xy2sn_quadratic . . . . .	5
<b>5</b>	<b>gis</b>	<b>6</b>
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
<b>6</b>	<b>shapefile/@Shp</b>	<b>6</b>
6.1	Shp . . . . .	6
6.2	area . . . . .	6
6.3	buffer . . . . .	6
6.4	cat . . . . .	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 . . . . .	7
6.11	cp . . . . .	7
6.12	create . . . . .	7

6.13	curvature . . . . .	8
6.14	cut . . . . .	8
6.15	diameter . . . . .	8
6.16	edges . . . . .	8
6.17	export_geo . . . . .	8
6.18	export_gpx . . . . .	8
6.19	export_gpx_track . . . . .	8
6.20	export_ldb . . . . .	8
6.21	export_poly . . . . .	8
6.22	export_sdf . . . . .	8
6.23	export_spline . . . . .	9
6.24	extract_coastline . . . . .	9
6.25	first_point . . . . .	9
6.26	flat . . . . .	9
6.27	generate_four_colour_index . . . . .	9
6.28	import_geo . . . . .	9
6.29	import_poly . . . . .	9
6.30	join_lines . . . . .	9
6.31	last_point . . . . .	9
6.32	length . . . . .	9
6.33	length2 . . . . .	10
6.34	line2point . . . . .	10
6.35	link_lines . . . . .	10
6.36	make_clockwise . . . . .	10
6.37	merge . . . . .	10
6.38	merge2 . . . . .	10
6.39	padd_nan . . . . .	10
6.40	plot . . . . .	10
6.41	points . . . . .	10
6.42	polygon_boundary . . . . .	10
6.43	read . . . . .	11
6.44	readZ . . . . .	11
6.45	remove_duplicate_points . . . . .	11
6.46	remove_leaves . . . . .	11
6.47	remove_nan . . . . .	11
6.48	remove_polygon_closure . . . . .	11
6.49	remove_short_elements . . . . .	11
6.50	renumber . . . . .	11
6.51	resample . . . . .	11
6.52	resample_2 . . . . .	11
6.53	resample_min . . . . .	12
6.54	resample_quick . . . . .	12
6.55	scale . . . . .	12
6.56	segment . . . . .	12

6.57	select_for_refinement . . . . .	12
6.58	set_geometry . . . . .	12
6.59	set_resolution . . . . .	12
6.60	skip . . . . .	12
6.61	smooth . . . . .	12
6.62	split_jump . . . . .	12
6.63	split_line . . . . .	13
6.64	split_nan . . . . .	13
6.65	swap_hemisphere . . . . .	13
6.66	translate . . . . .	13
6.67	write . . . . .	13
<b>7</b>	<b>shapefile</b>	<b>13</b>
7.1	astar_multi . . . . .	13
7.2	astar_recursive . . . . .	13
7.3	edge_chain . . . . .	13
7.4	edge_from_bnd . . . . .	13
7.5	preload_shp . . . . .	14
7.6	read_gpx . . . . .	14
7.7	shapewrite__ . . . . .	14
7.8	shapewrite_man . . . . .	16
7.9	shp2geo . . . . .	16
7.10	shp2kml . . . . .	16
7.11	shp_plot_attribute . . . . .	16
7.12	split_section . . . . .	16
7.13	write_polygon . . . . .	17

## 1 gis

### 1.1 GPX

### 1.2 batavia\_zero

## 2 centreline/@Centrelines

### 2.1 Centrelines

**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(mednan(z(kk))) meannan(z(kk)) min(z(kk)) max(z(kk)) ]
```

## 5.4 hgt\_read\_all

## 5.5 hgt\_resample

## 5.6 nmeatime

# 6 shapefile/@Shp

## 6.1 Shp

## 6.2 area

## 6.3 buffer

## 6.4 cat

## 6.5 clip

## 6.6 clip\_rect

## 6.7 close\_polygon

## 6.8 concat

## 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**   **cp**

**6.12**   **create**

**6.13**   **curvature**

**6.14**   **cut**

**6.15**   **diameter**

**6.16**   **edges**

**6.17**   **export\_geo**

**6.18**   **export\_gpx**

**6.19**   **export\_gpx\_track**

**6.20**   **export\_ldb**

6.21 `export_poly`

6.22 `export_sdf`

6.23 `export_spline`

6.24 `extract_coastline`

6.25 `first_point`

6.26 `flat`

6.27 `generate_four_colour_index`

6.28 `import_geo`

6.29 `import_poly`

6.30 `join_lines`

6.31 last\_point

6.32 length

6.33 length2

6.34 line2point

6.35 link\_lines

6.36 make\_clockwise

6.37 merge

6.38 merge2

6.39 padd\_nan

6.40 plot

6.41 points

6.42 polygon\_boundary

6.43 read

6.44 readZ

6.45 remove\_duplicate\_points

6.46 remove\_leaves

6.47 remove\_nan

6.48 remove\_polygon\_closure

6.49 remove\_short\_elements

6.50 renumber

**6.51    resample**

**6.52    resample\_2**

**6.53    resample\_min**

**6.54    resample\_quick**

**6.55    scale**

**6.56    segment**

**6.57    select\_for\_refinement**

**6.58    set\_geometry**

**6.59    set\_resolution**

**6.60    skip**

6.61 smooth

6.62 split\_jump

6.63 split\_line

6.64 split\_nan

6.65 swap\_hemisphere

6.66 translate

6.67 write

## 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 deftpfn  
 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`