

Package ‘GISSB’

July 10, 2026

Type Package

Title Network Analysis on the Norwegian Road Network

Version 1.1.1

Description A collection of GIS (Geographic Information System) functions in R, created for use in Statistics Norway. The functions are primarily related to network analysis on the Norwegian road network.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

Imports cppRouting, dplyr, here, httr, igraph, jsonlite, leaflet, magrittr, nabor, purrr, RCurl, reshape2, rlang, sf, stringr, tibble, tidygraph, tidyselect

VignetteBuilder knitr

URL <https://statisticsnorway.github.io/GISSB/>,
<https://github.com/statisticsnorway/GISSB>

BugReports <https://github.com/statisticsnorway/GISSB/issues>

Depends R (>= 3.5.0)

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Sindre Mikael Haugen [aut, cre]

Maintainer Sindre Mikael Haugen <sindre.haugen@ssb.no>

Repository CRAN

Date/Publication 2026-07-10 10:20:02 UTC

Contents

address_to_coords	2
coords_to_google	3
coords_to_node	3
edges_sampledata	5
graph_cppRouting_meters_sampledata	5
graph_cppRouting_minutes_sampledata	6
graph_sampledata	6
nodes_sampledata	7
path_leaflet	8
shortest_path_cppRouting	8
shortest_path_igraph	10
vegnett_sampledata	11
vegnett_to_R	12
Index	15

address_to_coords	<i>Convert addresses to coordinates</i>
-------------------	---

Description

The function address_to_coords can be used to find coordinates to supplied Norwegian addresses. Internet access is required as the function utilizes [the Norwegian Mapping Authority’s address API](#).

Usage

```
address_to_coords(zip_code, address, format = "sf", crs_out = 25833)
```

Arguments

zip_code	Character vector with zip codes.
address	Character vector with addresses (street name and house number).
format	Format of the returned object. Default value is set to “sf” (which returns an sf object). It is also possible set the format to “tibble” or “data.frame”.
crs_out	Numeric vector with the chosen coordinate reference system (CRS). Default value is set to CRS 25833.

Details

If no coordinates are found for the supplied address, the address could not be matched to an address registered in [Matrikkelen](#) – Norway’s official register of real property, buildings and addresses. You can search for registered addresses and properties in Kartverket’s [Eiendomsregisteret](#).

Value

Object with coordinates to the supplied addresses.

Examples

```
address_to_coords(zip_code = "0185", address = "Schweigaards gate 10")
```

coords_to_google	<i>Convert coordinates to Google Maps</i>
------------------	---

Description

The function `coords_to_google` can be used to convert coordinates of an `sf` object to a format that is easy to copy and paste into Google Maps.

Usage

```
coords_to_google(coords, crs_out = 25833)
```

Arguments

<code>coords</code>	An <code>sf</code> object with a geometry column that will be converted to CRS 4326.
<code>crs_out</code>	Chosen coordinate reference system (CRS) for the geometry column of the returned <code>sf</code> object.

Value

An `sf` object with a new column added (`coords_google`), and an additional `sf` geometry column with the chosen CRS.

Examples

```
address_to_coords(zip_code = "0185", address = "Schweigaards gate 10") %>%
  coords_to_google()
```

coords_to_node	<i>Connect coordinates to the nearest nodes in the road network</i>
----------------	---

Description

The function `coords_to_node` can be used to find the nearest nodes in the Norwegian road network (in meters) for chosen coordinates.

Usage

```
coords_to_node(
  coords,
  nodes_object = nodes,
  edges_object = edges,
  direction = "from",
  ID_col = "ID",
  crs_out = 25833,
  knn = 1,
  membership = FALSE
)
```

Arguments

coords	An sf object with the coordinates that should be connected to the road network.
nodes_object	An sf object with the nodes of the road network. This can be created with the function vegnett_to_R() .
edges_object	A data frame with the edges of the road network. This can be created with the function vegnett_to_R() .
direction	Character vector with from if the points should be from nodes or to if the points should be to nodes.
ID_col	Character vector with the name of the ID column. Default value is set to "ID".
crs_out	Numeric vector for the chosen coordinate reference system (CRS).
knn	Numeric vector with the chosen number of nodes that should be returned for each of the coordinates. If knn = 1 only the nearest nodes to the chosen coordinates will be returned. If knn = 2 the two nearest nodes will be returned etc.
membership	Logical. If TRUE the search for nodes is limited to nodes that belong to a road network that is connected either to the from or to nodes (only possible for either from or to). E.g. if you only want to search for from nodes that belong to the same road network as the to nodes, membership is set to FALSE in the search for to nodes and membership = TRUE for the from nodes (in that order).

Details

Before the function can be used, the nodes of the road network must be converted to an sf object that is called nodes (or another name supplied to the nodes_object argument). This can be done with the function [vegnett_to_R\(\)](#).

Value

An object (data.frame) with the following columns; from_nodeID/to_nodeID, membership_from_node/membership_to_node, coords_google_from_node/coords_google_to_node, knn_from_node/knn_to_node, and ID.

Examples

```
coords <- address_to_coords(zip_code = "0185", address = "Schweigaards gate 10")
coords_to_node(coords = coords,
```

```
direction = "from",
nodes_object = nodes_sampledata,
edges_object = edges_sampledata)
```

edges_sampledata	<i>Sample data (edges)</i>
------------------	----------------------------

Description

Sample of the Norwegian Road Network (vegnettRuteplan_FGDB_20210528.gdb) downloaded from Geonorge. Only the first 1000 rows from Oslo county are included.

Usage

```
data(edges_sampledata)
```

Format

A data frame

Source

National roads database - road network for routing

Examples

```
data(edges_sampledata)
class(edges_sampledata)
head(edges_sampledata)
```

graph_cppRouting_meters_sampledata	<i>Sample data (graph_cppRouting_meters)</i>
------------------------------------	--

Description

Sample of the Norwegian Road Network (vegnettRuteplan_FGDB_20210528.gdb) downloaded from Geonorge. Only the first 1000 rows from Oslo county are included.

Usage

```
data(graph_cppRouting_meters_sampledata)
```

Format

A cppRouting object

Source

National roads database - road network for routing

Examples

```
data(graph_cppRouting_meters_sampledata)
class(graph_cppRouting_meters_sampledata)
```

graph_cppRouting_minutes_sampledata	
	<i>Sample data (graph_cppRouting_minutes)</i>

Description

Sample of the Norwegian Road Network (vegnettRuteplan_FGDB_20210528.gdb) downloaded from Geonorge. Only the first 1000 rows from Oslo county are included.

Usage

```
data(graph_cppRouting_minutes_sampledata)
```

Format

A cppRouting object

Source

National roads database - road network for routing

Examples

```
data(graph_cppRouting_minutes_sampledata)
class(graph_cppRouting_minutes_sampledata)
```

graph_sampledata	<i>Sample data (graph)</i>
------------------	----------------------------

Description

Sample of the Norwegian Road Network (vegnettRuteplan_FGDB_20210528.gdb) downloaded from Geonorge. Only the first 1000 rows from Oslo county are included.

Usage

```
data(graph_sampledata)
```

Format

An igraph object.

Source

National roads database - road network for routing

Examples

```
data(graph_sampledata)
class(graph_sampledata)
```

nodes_sampledata	<i>Sample data (nodes)</i>
------------------	----------------------------

Description

Sample of the Norwegian Road Network (vegnettRuteplan_FGDB_20210528.gdb) downloaded from Geonorge. Only the first 1000 rows from Oslo county are included.

Usage

```
data(nodes_sampledata)
```

Format

An object of class sf.

Source

National roads database - road network for routing

Examples

```
data(nodes_sampledata)
class(nodes_sampledata)
head(nodes_sampledata)
```

path_leaflet	<i>Visualize the shortest path with Leaflet</i>
--------------	---

Description

The function `path_leaflet` visualizes the shortest path (in minutes or meters) that has been calculated with the function `shortest_path_igraph()` (where `path = TRUE`). Internet connection is required to load the background tiles.

Usage

```
path_leaflet(path, graph_object = graph)
```

Arguments

<code>path</code>	Object (list) that has been created with the function <code>shortest_path_igraph()</code> , where <code>path = TRUE</code> .
<code>graph_object</code>	The road network structured as a tidy graph (tbl_graph object). This can be done with the function <code>vegnett_to_R()</code> .

Value

Interactive Leaflet map that shows the shortest path (in minutes or meters) between a chosen from and to node in the road network.

Examples

```
shortest_path_igraph(from_node_ID = 25,
                     to_node_ID = 33,
                     unit = "minutes",
                     path = TRUE,
                     graph_object = graph_sampledata) %>%
  path_leaflet(graph_object = graph_sampledata)
```

shortest_path_cppRouting	<i>Shortest path (cppRouting)</i>
--------------------------	-----------------------------------

Description

The function `shortest_path_cppRouting` can be used to calculate the shortest path (either in minutes or meters) between two or more nodes in the Norwegian road network. The function also works with vectors with multiple from and to node ID's. Before the function can be used, the road network must be converted to a `cppRouting` object that is called `graph_cppRouting_minutes` or `graph_cppRouting_meters` (or other with other names supplied to the `graph_cppRouting_object` argument). This can be done with the function `vegnett_to_R()`.

Usage

```

shortest_path_cppRouting(
  from_node_ID,
  to_node_ID,
  unit = "minutes",
  dist = "all",
  graph_cppRouting_object = graph_cppRouting_minutes
)

```

Arguments

from_node_ID	Numeric vector with one or more from node ID's.
to_node_ID	Numeric vector with one or more to node ID's.
unit	Character vector with minutes to calculate the shortest path in minutes or meters for the shortest path in meters.
dist	Character vector that specifies if all the shortest paths between all the supplied from and to nodes are returned (all), or if only the minimum (min) or maximum (max) value for each from node ID is returned.
graph_cppRouting_object	The road network structured as a cppRouting graph object. This can be created with the function vegnett_to_R() .

Value

Object (data.frame) with how many minutes or meters the shortest path is between the supplied from and to node ID's.

Examples

```

shortest_path_cppRouting(from = 25,
  to = 33,
  unit = "minutes",
  graph_cppRouting_object = graph_cppRouting_minutes_sampledata)

shortest_path_cppRouting(from = 25,
  to = 33,
  unit = "meters",
  graph_cppRouting_object = graph_cppRouting_meters_sampledata)

shortest_path_cppRouting(from = 25,
  to = c(32, 33),
  unit = "minutes",
  dist = "min",
  graph_cppRouting_object = graph_cppRouting_minutes_sampledata)

```

shortest_path_igraph *Shortest path (igraph)*

Description

The function `shortest_path_igraph` can be used to calculate the shortest path (either in minutes or meters) between nodes in the Norwegian road network. The function can also return the node link, i.e. path, that the shortest path consists of.

Usage

```
shortest_path_igraph(
  from_node_ID,
  to_node_ID,
  graph_object = graph,
  unit = "minutes",
  path = FALSE
)
```

Arguments

<code>from_node_ID</code>	Numeric value with the from node ID (if multiple node ID's are to be used, see the function shortest_path_cppRouting()).
<code>to_node_ID</code>	Numeric value with the to node ID (if multiple node ID's are to be used, see the function shortest_path_cppRouting()).
<code>graph_object</code>	The road network structured as a tidy graph (<code>tbl_graph</code> object). This can be created with the function vegnett_to_R() .
<code>unit</code>	Character vector with <code>minutes</code> to calculate the shortest path in minutes or <code>meters</code> for the shortest path in meters.
<code>path</code>	Logical. If <code>TRUE</code> the node link with the shortest path is returned.

Details

Before the function can be used, the road network must be converted to a `tbl_graph` object. This can be done with the function [vegnett_to_R\(\)](#). There objects `graph` and `edges` needs to be loaded.

Value

Vector with the shortest path in minutes or meters. If `path = TRUE` the node link that the shortest path consists of is returned.

Examples

```
shortest_path_igraph(from_node_ID = 25,  
                     to_node_ID = 33,  
                     unit = "minutes",  
                     graph_object = graph_sampledata)  
  
shortest_path_igraph(from_node_ID = 25,  
                     to_node_ID = 33,  
                     unit = "meters",  
                     graph_object = graph_sampledata)  
  
shortest_path_igraph(from_node_ID = 25,  
                     to_node_ID = 33,  
                     unit = "minutes",  
                     path = TRUE,  
                     graph_object = graph_sampledata)
```

vegnett_sampledata	<i>Sample data (vegnett)</i>
--------------------	------------------------------

Description

Sample of the Norwegian Road Network (vegnettRuteplan_FGDB_20210528.gdb) downloaded from Geonorge. Only the first 1000 rows from Oslo county are included.

Usage

```
data(vegnett_sampledata)
```

Format

An object of class sf.

Source

[National roads database - road network for routing](#)

Examples

```
data(vegnett_sampledata)  
class(vegnett_sampledata)  
head(vegnett_sampledata)
```

vegnett_to_R	<i>Convert the Norwegian road network (NVDB Ruteplan nettverks-datasett) into network graphs in R</i>
--------------	---

Description

The function `vegnett_to_R` can be used to convert the Norwegian road network, downloaded from [Geonorge](#), to formats that enable network analysis in R (`tbl_graph` and `cppRouting`).

Usage

```
vegnett_to_R(
  vegnett,
  crs_out = 25833,
  year = 2022,
  fromnodeID = "FROMNODE",
  tonodeID = "TONODE",
  FT_minutes = "DRIVETIME_FW",
  TF_minutes = "DRIVETIME_BW",
  meters = "SHAPE_LENGTH",
  turn_restrictions = FALSE,
  ferry = TRUE
)
```

Arguments

<code>vegnett</code>	The Norwegian road network as an <code>sf</code> object, downloaded from Geonorge .
<code>crs_out</code>	Numeric vector with the chosen coordinate reference system (CRS). The default value is set to CRS 25833.
<code>year</code>	Numeric vector with the year the road network is from. Due to changes in the format of the files between 2021 and 2022, the most important thing is to choose between the "old" format (-2021) or the new format (2022-). The default value is set to 2022. Please see the example for the column names for 2021 and earlier.
<code>fromnodeID</code>	Character vector with the name of the column indicating the from node ID. Default value is set to FROMNODE (column name in 2022).
<code>tonodeID</code>	Character vector with the name of the column indicating the to node ID. Default value is set to TONODE (column name in 2022).
<code>FT_minutes</code>	Character vector with the name of the column that contains the cost in minutes from <code>fromnodeID</code> to <code>tonodeID</code> (FT). Default value is set to DRIVETIME_FW (column name in 2022).
<code>TF_minutes</code>	Character vector with the name of the column that contains the cost in minutes from <code>tonodeID</code> to <code>fromnodeID</code> (TF). Default value is set to DRIVETIME_BW (column name in 2022).
<code>meters</code>	Character vector with the name of the column that contains the cost in meters (equal for FT and TF). Default value is set to SHAPE_LENGTH (column name in 2022).

turn_restrictions

Logical. Default value is FALSE. If TRUE turn restrictions will be added to the road network. The turn restrictions layer from the road network file has to be loaded before this can be used (and the object has to be called `turnrestrictions_geom`). Due to errors in the turn restrictions file for 2022 it is not recommended to use this feature for now.

ferry

Logical/numeric vector. Default value is TRUE which means that all edges that involve ferries are given their original drive time (somewhere between 10 and 13 km/h). If a numeric value is supplied, the cost for all edges involving ferries will be converted to the supplied value in km/h.

Value

List containing the following elements:

- [1] `graph`: the road network structured as a tidy graph (`tbl_graph` object).
- [2] `nodes`: the road network's nodes (`sf` object).
- [3] `edges`: road network's edges/node links (`data.frame`).
- [4] `graph_cppRouting_minutes`: the road network structured as a `cppRouting` graph with the cost of travel in minutes (`cppRouting` object).
- [5] `graph_cppRouting_meters`: the road network structured as a `cppRouting` graph with the cost of travel in meters (`cppRouting` object).

Examples

```
vegnett_sampledata
vegnett_list <- vegnett_to_R(vegnett = vegnett_sampledata,
                           year = 2021,
                           fromnodeID = "FROMNODEID",
                           tonodeID = "TONODEID",
                           FT_minutes = "FT_MINUTES",
                           TF_minutes = "TF_MINUTES",
                           meters = "SHAPE_LENGTH")

graph <- vegnett_list[[1]]
nodes <- vegnett_list[[2]]
edges <- vegnett_list[[3]]
graph_cppRouting_minutes <- vegnett_list[[4]]
graph_cppRouting_meters <- vegnett_list[[5]]

graph
nodes
head(edges)
head(graph_cppRouting_minutes$data)
head(graph_cppRouting_minutes$coords)
head(graph_cppRouting_minutes$dict)
graph_cppRouting_minutes$nbnode

head(graph_cppRouting_meters$data)
head(graph_cppRouting_meters$coords)
head(graph_cppRouting_meters$dict)
graph_cppRouting_meters$nbnode
```


Index

* datasets

- edges_sampledata, [5](#)
- graph_cppRouting_meters_sampledata,
[5](#)
- graph_cppRouting_minutes_sampledata,
[6](#)
- graph_sampledata, [6](#)
- nodes_sampledata, [7](#)
- vegnett_sampledata, [11](#)

address_to_coords, [2](#)

coords_to_google, [3](#)

coords_to_node, [3](#)

edges_sampledata, [5](#)

graph_cppRouting_meters_sampledata, [5](#)

graph_cppRouting_minutes_sampledata, [6](#)

graph_sampledata, [6](#)

nodes_sampledata, [7](#)

path_leaflet, [8](#)

shortest_path_cppRouting, [8](#)

shortest_path_cppRouting(), [10](#)

shortest_path_igraph, [10](#)

shortest_path_igraph(), [8](#)

vegnett_sampledata, [11](#)

vegnett_to_R, [12](#)

vegnett_to_R(), [4](#), [8–10](#)