

# Package ‘jpmmap’

July 3, 2026

**Title** Japan Maps with Insets for Okinawa and Ogasawara

**Version** 0.1.3

**URL** <https://yhuriuchi.github.io/jpmmap/>,  
<https://github.com/yhuriuchi/jpmmap>

**BugReports** <https://github.com/yhuriuchi/jpmmap/issues>

**Description** Provides tools for drawing maps of Japan with prefecture and municipal boundaries. The plotting workflow mirrors the 'usmap' package and includes a transform that moves Okinawa and Ogasawara into visible inset locations. Boundary helpers build local 'GeoPackage' files from Japan's official MLIT N03 administrative area data  
<<https://nlftp.mlit.go.jp/ksj/gml/datalist/KsjTmplt-N03-2024.html>>.

**License** MIT + file LICENSE

**Copyright** See inst/COPYRIGHTS.

**Encoding** UTF-8

**Depends** R (>= 4.1.0)

**LazyData** true

**Imports** ggplot2, rlang, sf

**Suggests** dplyr, ggthemes, knitr, leaflet, pkgdown, rmarkdown, testthat (>= 3.0.0), tibble, tidyverse

**Config/testthat/edition** 3

**Config/Needs/website** pkgdown, knitr, rmarkdown, ggplot2, sf, leaflet, dplyr, ggthemes, tibble, tidyverse

**VignetteBuilder** knitr

**RoxygenNote** 7.3.2

**NeedsCompilation** no

**Author** Yusaku Horiuchi [aut, cre]

**Maintainer** Yusaku Horiuchi <yusaku.horiuchi@gmail.com>

**Repository** CRAN

**Date/Publication** 2026-07-03 12:00:02 UTC

## Contents

jpmmap_crs . . . . .	2
jpmmap_data . . . . .	2
jpmmap_transform . . . . .	4
jp_disputed_territories . . . . .	5
jp_map . . . . .	6
jp_map_join . . . . .	7
jp_map_leaflet . . . . .	8
jp_map_with_data . . . . .	11
jp_prefecture_gdp . . . . .	11
jp_us_military_bases . . . . .	12
plot_jpmmap . . . . .	13
<b>Index</b>	<b>16</b>

---

jpmmap_crs	<i>Coordinate Reference System for jpmmap</i>
------------	---

---

### Description

Returns the projected coordinate reference system used by jpmmap.

### Usage

```
jpmmap_crs()
```

### Value

An `sf::st_crs()` object.

---

jpmmap_data	<i>Manage jpmmap Boundary Data</i>
-------------	------------------------------------

---

### Description

Helpers for locating and building the GeoPackage boundary data used by jpmmap. Boundary GeoPackage files can come from the separate `jpmmapdata` package or from files built locally with `jpmmap_build_data()`.

**Usage**

```

jpmmap_data_dir(create = TRUE)

available_jpmmap_data(data_dir = NULL)

jpmmap_build_data(
  year = 2024,
  prefecture = NULL,
  destdir = jpmmap_data_dir(),
  url = NULL,
  overwrite = FALSE,
  quiet = FALSE,
  simplify_tolerance = NULL
)

```

**Arguments**

create	Whether to create the default data directory.
data_dir	Optional directory to scan for boundary data.
year	Boundary data year.
prefecture	Optional prefecture code, English name, or Japanese name. When supplied, only that prefecture's official MLIT N03 file is downloaded and built.
destdir	Directory where the generated GeoPackage should be written.
url	Optional source URL. By default, an MLIT N03 URL is constructed.
overwrite	Whether to overwrite an existing GeoPackage.
quiet	Whether to suppress messages from download and spatial reads/writes.
simplify_tolerance	Optional tolerance passed to <code>sf::st_simplify()</code> .

**Value**

`jpmmap_data_dir()` returns a path, `available_jpmmap_data()` returns a data frame with `year`, `pref_code`, `prefecture`, `source`, and `path`, and `jpmmap_build_data()` invisibly returns the generated file.

**Source**

MLIT National Land Numerical Information N03 administrative area data: <https://nlftp.mlit.go.jp/ksj/gml/datalist/KsjTmplt-N03-2024.html>.

**Examples**

```

jpmmap_data_dir(create = FALSE)
available_jpmmap_data()

jpmmap_build_data(

```

```

year = 2024,
prefecture = "Ehime",
destdir = tempdir(),
overwrite = TRUE,
quiet = TRUE
)

```

---

jpmmap\_transform

*Transform Data to the jpmmap Projection*


---

### Description

Transforms geographic data to the jpmmap projected coordinate system. When `inset = TRUE`, Okinawa and Ogasawara are moved to visible inset locations. Use `okinawa = FALSE`, `ogasawara = FALSE`, `inset = "okinawa"`, or `inset = "ogasawara"` to transport only one island group.

### Usage

```

jpmmap_transform(
  data,
  input_names = c("lon", "lat"),
  output_names = input_names,
  inset = TRUE,
  okinawa = TRUE,
  ogasawara = TRUE
)

```

### Arguments

<code>data</code>	An sf object, sfc geometry vector, or data frame.
<code>input_names</code>	Longitude and latitude column names for data frames.
<code>output_names</code>	Output coordinate column names for data frames.
<code>inset</code>	Inset behavior. Use TRUE to move both Okinawa and Ogasawara, FALSE for no movement, or a character vector containing "okinawa" and/or "ogasawara" to move selected island groups.
<code>okinawa</code>	Whether Okinawa should be moved when inset includes it.
<code>ogasawara</code>	Whether Ogasawara should be moved when inset includes it.

### Value

An object of the same general type as `data`.

**Examples**

```

if (requireNamespace("tibble", quietly = TRUE)) {
  places <- tibble::tribble(
    ~place, ~lon, ~lat,
    "Tokyo", 139.767, 35.681,
    "Naha", 127.681, 26.212,
    "Ogasawara", 142.191, 27.094
  )

  places |>
    jpmmap_transform(output_names = c("x", "y"))
  places |>
    jpmmap_transform(output_names = c("x", "y"), inset = "okinawa")
  places |>
    jpmmap_transform(output_names = c("x", "y"), ogasawara = FALSE)
}

```

---

jp\_disputed\_territories

*Retrieve Disputed-Territory Island And Reef Shapes*


---

**Description**

Returns cartographic island/reef shapes for disputed territories or disputed maritime/EEZ-status areas discussed in Japan territorial-dispute references. These geometries are intentionally separate from the MLIT N03 administrative boundary data and are not authoritative legal boundary polygons.

**Usage**

```

jp_disputed_territories(
  territorial_disputes = TRUE,
  regions = c("prefectures", "prefecture", "municipalities", "municipality"),
  inset = TRUE,
  okinawa = TRUE,
  ogasawara = TRUE
)

```

**Arguments**

territorial_disputes	Which disputed-territory shapes to include. Use TRUE for all built-in shapes, FALSE for none, or a character vector containing one or more of "northern_territories", "okinotorishima", "senkaku", and "takeshima". Common aliases such as "kuril", "liancourt", "dokdo", and "diaoyu" are also accepted.
regions	Boundary level whose columns should be mirrored: prefectures or municipalities.
inset	Inset behavior. Use TRUE to move Okinawa and Ogasawara, FALSE for literal projected locations, or a character vector containing "okinawa" and/or "ogasawara".

okinawa	Whether Okinawa should be moved when inset includes it.
ogasawara	Whether Ogasawara should be moved when inset includes it.

**Value**

An sf data frame.

**Source**

Territory list based on [https://en.wikipedia.org/wiki/Territorial\\_disputes\\_of\\_Japan](https://en.wikipedia.org/wiki/Territorial_disputes_of_Japan).  
Shapes are derived from Natural Earth 1:10m land polygons and OpenStreetMap geometry.

**Examples**

```
jp_disputed_territories()
jp_disputed_territories(c("senkaku", "takeshima"))
```

---

jp_map	<i>Retrieve Japan Map Data</i>
--------	--------------------------------

---

**Description**

Reads Japan administrative boundary data and returns an sf object.

**Usage**

```
jp_map(
  regions = c("prefectures", "prefecture", "municipalities", "municipality"),
  include = c(),
  exclude = c(),
  data_year = NULL,
  inset = TRUE,
  okinawa = TRUE,
  ogasawara = TRUE,
  territorial_disputes = TRUE,
  data_dir = NULL
)
```

**Arguments**

regions	Boundary level: prefectures or municipalities.
include	Regions to include by code, English name, or Japanese name.
exclude	Regions to exclude by code, English name, or Japanese name.
data_year	Boundary data year. The newest appropriate available file is used by default. For example, national prefecture maps prefer a national file, while one-prefecture municipal requests can use a matching prefecture file.

inset	Inset behavior. Use TRUE to move both Okinawa and Ogasawara, FALSE for no movement, or a character vector containing "okinawa" and/or "ogasawara" to move selected island groups.
okinawa	Whether Okinawa should be moved when inset includes it.
ogasawara	Whether Ogasawara should be moved when inset includes it.
territorial_disputes	Whether to include disputed-territory island/reef shapes. Use FALSE to exclude them, or a character vector containing one or more of "northern_territories", "okinotorishima", "senkaku", and "takeshima".
data_dir	Optional directory containing jpmmap_boundaries_YYYY.gpkg.

**Value**

An sf data frame.

**Examples**

```
if (nrow(available_jpmmap_data()) > 0) {
  jp_map("prefectures")
  jp_map("prefectures", okinawa = FALSE)
  jp_map("prefectures", territorial_disputes = FALSE)
  jp_map("municipalities", include = "Okinawa")
  jp_map("prefecture")
  jp_map("municipality", include = "Okinawa")
}
```

---

 jp\_map\_join

*Join User Data to a jpmmap Boundary Object*


---

**Description**

jp\_map\_join() attaches columns from a user data frame to an sf object returned by jp\_map(). It is designed for common Japan-map keys, including pref\_code, jis\_code, and municipality\_code, and keeps leading zeroes from getting in the user's way.

**Usage**

```
jp_map_join(
  map,
  data,
  by = NULL,
  values = NULL,
  unmatched = c("warning", "error", "ignore"),
  multiple = c("error", "first")
)
```

**Arguments**

map	An sf object returned by <code>jp_map()</code> .
data	A data frame with one row per region to join.
by	Join column. If NULL, <code>jpmap</code> guesses from common columns such as <code>jis_code</code> , <code>municipality_code</code> , <code>pref_code</code> , <code>prefecture</code> , <code>prefecture_ja</code> , <code>municipality</code> , and <code>municipality_ja</code> . To join columns with different names, use a named character vector such as <code>c("pref_code" = "code")</code> , where the name is the map column and the value is the data column.
values	Optional column expected to exist after joining. This is useful when checking data before passing it to a plotting function.
unmatched	What to do when user data rows do not match the map, or map regions do not receive a data row. One of "warning", "error", or "ignore".
multiple	What to do when data has duplicate join keys. One of "error" or "first".

**Value**

An sf object with non-geometry columns from data joined to map.

**Examples**

```
if (requireNamespace("dplyr", quietly = TRUE) &&
    nrow(available_jpmap_data()) > 0) {
  data("jp_prefecture_gdp")

  gdp <- jp_prefecture_gdp |>
    dplyr::select(pref_code, gdp_per_capita_jpy)

  joined <- jp_map("prefecture") |>
    jp_map_join(gdp, by = "pref_code")

  "gdp_per_capita_jpy" %in% names(joined)
}
```

---

 jp\_map\_leaflet

*Draw an Interactive Japan Map with leaflet*


---

**Description**

`jp_map_leaflet()` creates a simple interactive web map from `jpmap` boundaries. It uses literal longitude/latitude geography because Leaflet web tiles expect WGS84 coordinates. Use `plot_jpmap()` when you want the static Okinawa and Ogasawara inset layout.

**Usage**

```

jp_map_leaflet(
  regions = c("prefectures", "prefecture", "municipalities", "municipality"),
  include = c(),
  exclude = c(),
  data = data.frame(),
  values = NULL,
  by = NULL,
  data_year = NULL,
  territorial_disputes = TRUE,
  data_dir = NULL,
  palette = "Blues",
  fill = "grey92",
  color = "grey35",
  weight = 1,
  opacity = 1,
  fill_opacity = 0.75,
  na_color = "#D9D9D9",
  label = NULL,
  popup = NULL,
  tiles = TRUE,
  legend = TRUE,
  fit_bounds = TRUE,
  simplify_tolerance = NULL,
  disputed_fill = NULL,
  disputed_color = NULL,
  disputed_dots = FALSE,
  disputed_dot_radius = 5,
  ...
)

```

**Arguments**

<code>regions</code>	Boundary level: prefectures or municipalities.
<code>include</code>	Regions to include by code, English name, or Japanese name.
<code>exclude</code>	Regions to exclude by code, English name, or Japanese name.
<code>data</code>	Optional data frame to join to the map before drawing.
<code>values</code>	Optional column to use for polygon fill colors.
<code>by</code>	Optional join column passed to <code>jp_map_join()</code> .
<code>data_year</code>	Boundary data year. The newest appropriate available file is used by default.
<code>territorial_disputes</code>	Whether to include disputed-territory island/reef shapes. Use <code>FALSE</code> to exclude them, or a selected character vector.
<code>data_dir</code>	Optional directory containing <code>jpmap_boundaries_YYYY.gpkg</code> .
<code>palette</code>	Palette name or color vector passed to Leaflet palette functions when values is supplied.

fill	Polygon fill color used when values is NULL.
color	Polygon outline color.
weight	Polygon outline weight.
opacity	Polygon outline opacity.
fill_opacity	Polygon fill opacity.
na_color	Fill color for missing values.
label	NULL for default region labels, FALSE for no labels, a column name, or a character vector with one value per map row.
popup	NULL for default popups, FALSE for no popups, a column name, or a character vector with one value per map row.
tiles	Whether to add default OpenStreetMap tiles.
legend	Whether to add a legend when values is supplied.
fit_bounds	Whether to zoom the widget to the map bounds.
simplify_tolerance	Optional tolerance passed to <code>sf::st_simplify()</code> before drawing polygons. This is useful for smaller website widgets.
disputed_fill	Optional fill color for disputed-territory shapes. When NULL, the ordinary map fill is used.
disputed_color	Optional outline color for disputed-territory shapes. When NULL, the ordinary map outline is used.
disputed_dots	Whether to draw circle markers on disputed-territory shapes.
disputed_dot_radius	Radius for disputed-territory circle markers.
...	Additional arguments passed to <code>leaflet::leaflet()</code> .

**Value**

A leaflet `htmlwidget`.

**Examples**

```
if (requireNamespace("leaflet", quietly = TRUE) &&
    requireNamespace("dplyr", quietly = TRUE) &&
    nrow(available_jpmap_data()) > 0) {
  data("jp_prefecture_gdp")

  gdp <- jp_prefecture_gdp |>
    dplyr::select(pref_code, prefecture, gdp_per_capita_jpy)

  jp_map_leaflet(
    "prefecture",
    data = gdp,
    values = "gdp_per_capita_jpy",
    popup = "prefecture"
  )
}
```

---

jp\_map\_with\_data      *Join Data to a jpmmap Map*

---

### Description

Joins user data to a Japan map object. This is a compact wrapper around `jp_map_join()` kept for plotting workflows that call it internally.

### Usage

```
jp_map_with_data(map, data, values = NULL, by = NULL)
```

### Arguments

<code>map</code>	An sf object returned by <code>jp_map()</code> .
<code>data</code>	A data frame containing a matching administrative code or name column.
<code>values</code>	Optional value column to check after joining.
<code>by</code>	Optional join column. If omitted, <code>jpmmap</code> guesses from common columns.

### Value

An sf object.

---

jp\_prefecture\_gdp      *GDP per capita by Japanese prefecture*

---

### Description

A sample data frame containing 2021 prefecture GDP per capita values.

### Format

A data frame with 47 rows and 6 variables:

**pref\_code** Two-digit Japanese prefecture code.  
**prefecture** Prefecture name in English.  
**year** Reference year.  
**gdp\_per\_capita\_jpy** GDP per capita in Japanese yen.  
**gdp\_per\_capita\_usd\_ppp** GDP per capita in U.S. dollars at PPP.  
**source** Source note.

### Source

OECD 2021 regional GDP per-capita values as tabulated at [https://en.wikipedia.org/wiki/List\\_of\\_Japanese\\_prefectures\\_by\\_GDP\\_per\\_capita](https://en.wikipedia.org/wiki/List_of_Japanese_prefectures_by_GDP_per_capita).

---

jp\_us\_military\_bases *Selected U.S. military installations in Japan*

---

### Description

A sample data frame containing selected U.S. military installations in Japan with longitude/latitude coordinates and public approximate personnel figures where those figures have a clear source.

### Format

A data frame with 17 rows and 12 variables:

**base** Installation or command name.

**branch** Primary U.S. military branch.

**prefecture** Japanese prefecture.

**municipality** Municipality or municipalities.

**lon** Longitude in WGS84.

**lat** Latitude in WGS84.

**personnel** Approximate public personnel figure, when available.

**personnel\_scope** What the public figure counts.

**personnel\_geography** Whether the figure describes an installation, regional total, command, or broader installation community.

**personnel\_is\_base\_specific** Whether personnel is interpreted as a base-specific figure.

**source\_url** Source URL.

**note** Caveat for use in examples.

### Details

These values are intended for examples and are not an official personnel accounting. Some rows describe broader installation, regional, command, or community populations; see `personnel_scope` and `personnel_geography`.

### Source

Public installation, command, and regional pages listed in the `source_url` column, including [https://en.wikipedia.org/wiki/Misawa\\_Air\\_Base](https://en.wikipedia.org/wiki/Misawa_Air_Base), [https://en.wikipedia.org/wiki/Yokota\\_Air\\_Base](https://en.wikipedia.org/wiki/Yokota_Air_Base), and [https://en.wikipedia.org/wiki/Okinawa\\_Island](https://en.wikipedia.org/wiki/Okinawa_Island).

---

`plot_jpmap`*Plot a Japan Map*

---

**Description**

Plots Japan prefecture or municipal boundaries using ggplot2.

**Usage**

```
plot_jpmap(  
  regions = c("prefectures", "prefecture", "municipalities", "municipality"),  
  include = c(),  
  exclude = c(),  
  data = data.frame(),  
  values = NULL,  
  labels = FALSE,  
  label_color = "black",  
  data_year = NULL,  
  inset = TRUE,  
  okinawa = TRUE,  
  ogasawara = TRUE,  
  territorial_disputes = TRUE,  
  disputed_fill = NULL,  
  disputed_color = NULL,  
  disputed_linewidth = NULL,  
  disputed_dots = FALSE,  
  disputed_dot_fill = "#001040",  
  disputed_dot_color = "white",  
  disputed_dot_size = 1.25,  
  disputed_dot_stroke = 0.2,  
  inset_boxes = TRUE,  
  inset_box_color = "grey50",  
  inset_box_linewidth = 0.35,  
  data_dir = NULL,  
  xlim = NULL,  
  ylim = NULL,  
  x_breaks = ggplot2::waiver(),  
  y_breaks = ggplot2::waiver(),  
  x_labels = ggplot2::waiver(),  
  y_labels = ggplot2::waiver(),  
  fill = "grey92",  
  color = "grey35",  
  linewidth = 0.25,  
  ...  
)
```

**Arguments**

regions	Boundary level: prefectures or municipalities.
include	Regions to include by code, English name, or Japanese name.
exclude	Regions to exclude by code, English name, or Japanese name.
data	Optional data frame to join to the map.
values	Column name in data to use as the fill variable.
labels	Whether to draw region labels.
label_color	Label text color.
data_year	Boundary data year.
inset	Inset behavior. Use TRUE to move both Okinawa and Ogasawara, FALSE for no movement, or a character vector containing "okinawa" and/or "ogasawara" to move selected island groups. For filtered municipality maps such as <code>plot_jpmap("municipality", include = "Okinawa")</code> , the default plot frame is local and non-inset unless inset is supplied explicitly.
okinawa	Whether Okinawa should be moved when inset includes it.
ogasawara	Whether Ogasawara should be moved when inset includes it.
territorial_disputes	Whether to include disputed-territory island/reef shapes. Use FALSE to exclude them, or a character vector containing one or more of "northern_territories", "okinotorishima", "senkaku", and "takeshima".
disputed_fill	Optional fill color for disputed-territory shapes. When NULL, the ordinary map fill is used.
disputed_color	Optional outline color for disputed-territory shapes. When NULL, the ordinary map outline is used.
disputed_linewidth	Optional line width for disputed-territory shapes.
disputed_dots	Whether to draw dot markers on disputed-territory shapes.
disputed_dot_fill	Fill color for disputed-territory dot markers.
disputed_dot_color	Outline color for disputed-territory dot markers.
disputed_dot_size	Size for disputed-territory dot markers.
disputed_dot_stroke	Stroke width for disputed-territory dot markers.
inset_boxes	Whether to draw boxes around transported Okinawa and Ogasawara insets.
inset_box_color	Outline color for inset boxes.
inset_box_linewidth	Line width for inset boxes.
data_dir	Optional directory containing <code>jpmap_boundaries_YYYY.gpkg</code> .

<code>xlim, ylim</code>	Optional longitude and latitude limits for the plot frame.
<code>x_breaks, y_breaks</code>	Optional longitude and latitude axis breaks.
<code>x_labels, y_labels</code>	Optional longitude and latitude axis labels.
<code>fill</code>	Boundary fill color when values is not supplied.
<code>color</code>	Boundary line color.
<code>linewidth</code>	Boundary line width.
<code>...</code>	Additional arguments passed to <code>ggplot2::geom_sf()</code> .

**Value**

A `ggplot2` plot.

**Examples**

```
if (nrow(available_jpmap_data()) > 0) {  
  plot_jpmap("prefecture")  
  plot_jpmap("prefecture", ogasawara = FALSE)  
  plot_jpmap("prefecture", territorial_disputes = FALSE)  
  plot_jpmap(  
    "prefecture",  
    ogasawara = FALSE,  
    xlim = c(122, 149),  
    ylim = c(28.5, 47),  
    x_breaks = seq(125, 145, 5),  
    y_breaks = seq(30, 45, 5)  
  )  
  plot_jpmap("prefecture", inset_boxes = FALSE)  
  plot_jpmap("municipality", include = "Okinawa")  
}
```

# Index

## \* datasets

- jp\_prefecture\_gdp, [11](#)
- jp\_us\_military\_bases, [12](#)

available\_jpmap\_data (jpmap\_data), [2](#)

jp\_disputed\_territories, [5](#)

jp\_map, [6](#)

jp\_map\_join, [7](#)

jp\_map\_leaflet, [8](#)

jp\_map\_with\_data, [11](#)

jp\_prefecture\_gdp, [11](#)

jp\_us\_military\_bases, [12](#)

jpmap\_build\_data (jpmap\_data), [2](#)

jpmap\_crs, [2](#)

jpmap\_data, [2](#)

jpmap\_data\_dir (jpmap\_data), [2](#)

jpmap\_transform, [4](#)

plot\_jpmap, [13](#)