2022-04-25

Contents

Chapter 1

RStudio `~/Lab_Data/`

This is a *sample* book written in **Markdown**. You can use anything that Pandoc's Markdown supports; for example, a math equation $a^2 + b^2 = c^2$.

1.1 Usage

Each **bookdown** chapter is an .Rmd file, and each .Rmd file can contain one (and only one) chapter. A chapter *must* start with a first-level heading: `# A good chapter`, and can contain one (and only one) first-level heading.

Use second-level and higher headings within chapters like: `## A short section` or `### An even shorter section`.

The `index.Rmd` file is required, and is also your first book chapter. It will be the homepage when you render the book.

1.2 Render book

You can render the HTML version of this example book without changing anything:

1. Find the **Build** pane in the RStudio IDE, and
2. Click on **Build Book**, then select your output format, or select "All formats" if you'd like to use multiple formats from the same book source files.

Or build the book from the R console:

```
bookdown::render_book()
```

To render this example to PDF as a `bookdown::pdf_book`, you'll need to install XeLaTeX. You are recommended to install TinyTeX (which includes XeLaTeX): https://yihui.org/tinytex/.

1.3 Preview book

As you work, you may start a local server to live preview this HTML book. This preview will update as you edit the book when you save individual .Rmd files. You can start the server in a work session by using the RStudio add-in "Preview book", or from the R console:

```
bookdown::serve_book()
```

Chapter 2

Hello bookdown

All chapters start with a first-level heading followed by your chapter title, like the line above. There should be only one first-level heading (`#`) per .Rmd file.

2.1 A section

All chapter sections start with a second-level (`##`) or higher heading followed by your section title, like the sections above and below here. You can have as many as you want within a chapter.

An unnumbered section

Chapters and sections are numbered by default. To un-number a heading, add a `{.unnumbered}` or the shorter `{-}` at the end of the heading, like in this section.

Chapter 3

Cross-references

Cross-references make it easier for your readers to find and link to elements in your book.

3.1 Chapters and sub-chapters

There are two steps to cross-reference any heading:

1. Label the heading: `# Hello world {#nice-label}`.
    - Leave the label off if you like the automated heading generated based on your heading title: for example, `# Hello world` = `# Hello world {#hello-world}`.
    - To label an un-numbered heading, use: `# Hello world {-#nice-label}` or `{# Hello world .unnumbered}`.
2. Next, reference the labeled heading anywhere in the text using `\@ref(nice-label)`; for example, please see Chapter 3.
    - If you prefer text as the link instead of a numbered reference use: any text you want can go here.

3.2 Captioned figures and tables

Figures and tables *with captions* can also be cross-referenced from elsewhere in your book using `\@ref(fig:chunk-label)` and `\@ref(tab:chunk-label)`, respectively.

See Figure 3.1.

```
par(mar = c(4, 4, .1, .1))
plot(pressure, type = 'b', pch = 19)
```

Don't miss Table 3.1.

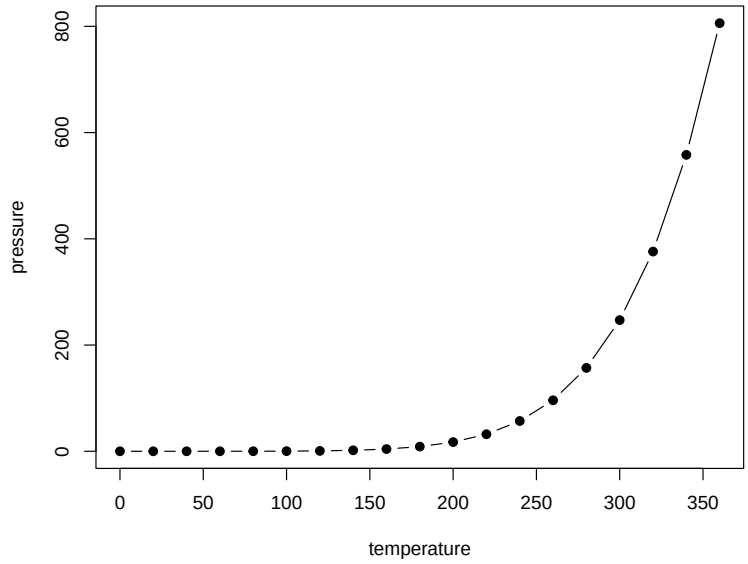


Figure 3.1: Here is a nice figure!

```
knitr::kable(
  head(pressure, 10), caption = 'Here is a nice table!',
  booktabs = TRUE
)
```

Table 3.1: Here is a nice table!

temperature	pressure
0	0.0002
20	0.0012
40	0.0060
60	0.0300
80	0.0900
100	0.2700
120	0.7500
140	1.8500
160	4.2000
180	8.8000

Chapter 4

Parts

You can add parts to organize one or more book chapters together. Parts can be inserted at the top of an .Rmd file, before the first-level chapter heading in that same file.

Add a numbered part: `# (PART) Act one {-}` (followed by `# A chapter`)

Add an unnumbered part: `# (PART\*) Act one {-}` (followed by `# A chapter`)

Add an appendix as a special kind of un-numbered part: `# (APPENDIX) Other stuff {-}` (followed by `# A chapter`). Chapters in an appendix are prepended with letters instead of numbers.

Chapter 5

Footnotes and citations

5.1 Footnotes

Footnotes are put inside the square brackets after a caret `^[]`. Like this one [1].

5.2 Citations

Reference items in your bibliography file(s) using `@key`.

For example, we are using the **bookdown** package (Xie, 2021) (check out the last code chunk in index.Rmd to see how this citation key was added) in this sample book, which was built on top of R Markdown and **knitr** (Xie, 2015) (this citation was added manually in an external file book.bib). Note that the `.bib` files need to be listed in the index.Rmd with the YAML `bibliography` key.

The `bs4_book` theme makes footnotes appear inline when you click on them. In this example book, we added `csl: chicago-fullnote-bibliography.csl` to the `index.Rmd` YAML, and include the `.csl` file. To download a new style, we recommend: https://www.zotero.org/styles/

The RStudio Visual Markdown Editor can also make it easier to insert citations: https://rstudio.github.io/visual-markdown-editing/#/citations

[1] This is a footnote.

Chapter 6

Blocks

6.1 Equations

Here is an equation.

$$f(k) = \binom{n}{k} p^k (1-p)^{n-k} \tag{6.1}$$

You may refer to using `\@ref(eq:binom)`, like see Equation (6.1).

6.2 Theorems and proofs

Labeled theorems can be referenced in text using `\@ref(thm:tri)`, for example, check out this smart theorem 6.1.

Theorem 6.1. *For a right triangle, if c denotes the* length *of the hypotenuse and a and b denote the lengths of the* ***other*** *two sides, we have*

$$a^2 + b^2 = c^2$$

Read more here https://bookdown.org/yihui/bookdown/markdown-extensions-by-bookdown.html.

6.3 Callout blocks

The `bs4_book` theme also includes special callout blocks, like this `.rmdnote`.

You can use **markdown** inside a block.

```
head(beaver1, n = 5)
#>   day time  temp activ
#> 1 346  840 36.33     0
#> 2 346  850 36.34     0
#> 3 346  900 36.35     0
#> 4 346  910 36.42     0
#> 5 346  920 36.55     0
```

It is up to the user to define the appearance of these blocks for LaTeX output.

You may also use: `.rmdcaution`, `.rmdimportant`, `.rmdtip`, or `.rmdwarning` as the block name.

The R Markdown Cookbook provides more help on how to use custom blocks to design your own callouts: https://bookdown.org/yihui/rmarkdown-cookbook/custom-blocks.html

Chapter 7

Sharing your book

7.1 Publishing

HTML books can be published online, see: https://bookdown.org/yihui/bookdown/publishing.html

7.2 404 pages

By default, users will be directed to a 404 page if they try to access a webpage that cannot be found. If you'd like to customize your 404 page instead of using the default, you may add either a `_404.Rmd` or `_404.md` file to your project root and use code and/or Markdown syntax.

7.3 Metadata for sharing

Bookdown HTML books will provide HTML metadata for social sharing on platforms like Twitter, Facebook, and LinkedIn, using information you provide in the `index.Rmd` YAML. To setup, set the `url` for your book and the path to your `cover-image` file. Your book's `title` and `description` are also used.

This `bs4_book` provides enhanced metadata for social sharing, so that each chapter shared will have a unique description, auto-generated based on the content.

Specify your book's source repository on GitHub as the `repo` in the `_output.yml` file, which allows users to view each chapter's source file or suggest an edit. Read more about the features of this output format here:

https://pkgs.rstudio.com/bookdown/reference/bs4_book.html

Or use:

```
?bookdown::bs4_book
```

Chapter 8

```
library(tidyverse)  # Essential package
#> -- Attaching packages ------------------- tidyverse 1.3.1 --
#> v ggplot2 3.3.5     v purrr   0.3.4
#> v tibble  3.1.6     v dplyr   1.0.8
#> v tidyr   1.2.0     v stringr 1.4.0
#> v readr   2.1.2     v forcats 0.5.1
#> -- Conflicts ---------------------- tidyverse_conflicts() --
#> x dplyr::filter() masks stats::filter()
#> x dplyr::lag()    masks stats::lag()
library(ggpubr)     # Publication-oriented figures
library(kableExtra) # Tables
#>
#> Attaching package: 'kableExtra'
#> The following object is masked from 'package:dplyr':
#>
#>     group_rows
library(magick)     # Imagemagick R API
#> Linking to ImageMagick 6.9.11.60
#> Enabled features: fontconfig, freetype, fftw, heic, lcms, pango, webp, x11
#> Disabled features: cairo, ghostscript, raw, rsvg
#> Using 32 threads
library(patchwork)  # Simplified figure tiling
library(ggspatial)  # Essential for map-making with ggplot
library(sf)         # Essential for map data manipulation
#> Linking to GEOS 3.9.0, GDAL 3.2.2, PROJ 7.2.1; sf_use_s2() is TRUE
library(showtext)   # I want to use google fonts in the figures
#> Loading required package: sysfonts
#> Loading required package: showtextdb
```

Noto Sans Google Fonts

```
font_add_google("Noto Sans","notosans")
```

ggplot theme_pubr() ggplot

```
theme_pubr(base_size = 10, base_family = "notosans") |> theme_set()
showtext_auto() # Automatically embed the Noto Sans fonts into the ggplots.
```

8.1

(shapefile) shp, shx, dbf prj xml

~/Lab_Data/Japan_map_data/Japan

```
mlit = read_sf("~/Lab_Data/Japan_map_data/Japan/N03-20210101_GML/")
```

```
mlit
```

(CRS: Coordinate Reference System)

```
st_crs(mlit)
#> Coordinate Reference System:
#>   User input: JGD2011
#>   wkt:
#> GEOGCRS["JGD2011",
#>     DATUM["Japanese Geodetic Datum 2011",
#>         ELLIPSOID["GRS 1980",6378137,298.257222101,
#>             LENGTHUNIT["metre",1]]],
#>     PRIMEM["Greenwich",0,
#>         ANGLEUNIT["degree",0.0174532925199433]],
#>     CS[ellipsoidal,2],
#>         AXIS["geodetic latitude (Lat)",north,
#>             ORDER[1],
#>             ANGLEUNIT["degree",0.0174532925199433]],
#>         AXIS["geodetic longitude (Lon)",east,
#>             ORDER[2],
#>             ANGLEUNIT["degree",0.0174532925199433]],
#>     USAGE[
#>         SCOPE["Horizontal component of 3D system."],
#>         AREA["Japan - onshore and offshore."],
#>         BBOX[17.09,122.38,46.05,157.65]],
#>     ID["EPSG",6668]]
```

CRS 2 EPSG

```
# HTML
tibble(`EPSG Code` = c(4326,6668,6677),
       `CRS` = c("WGS84", "JGD2011", "JGD2011 / Japan Plane Rectangular CS IX"),
```

EPSG Code	CRS	Units
4326	WGS84	degrees
6668	JGD2011	degrees
6677	JGD2011 / Japan Plane Rectangular CS IX	meters

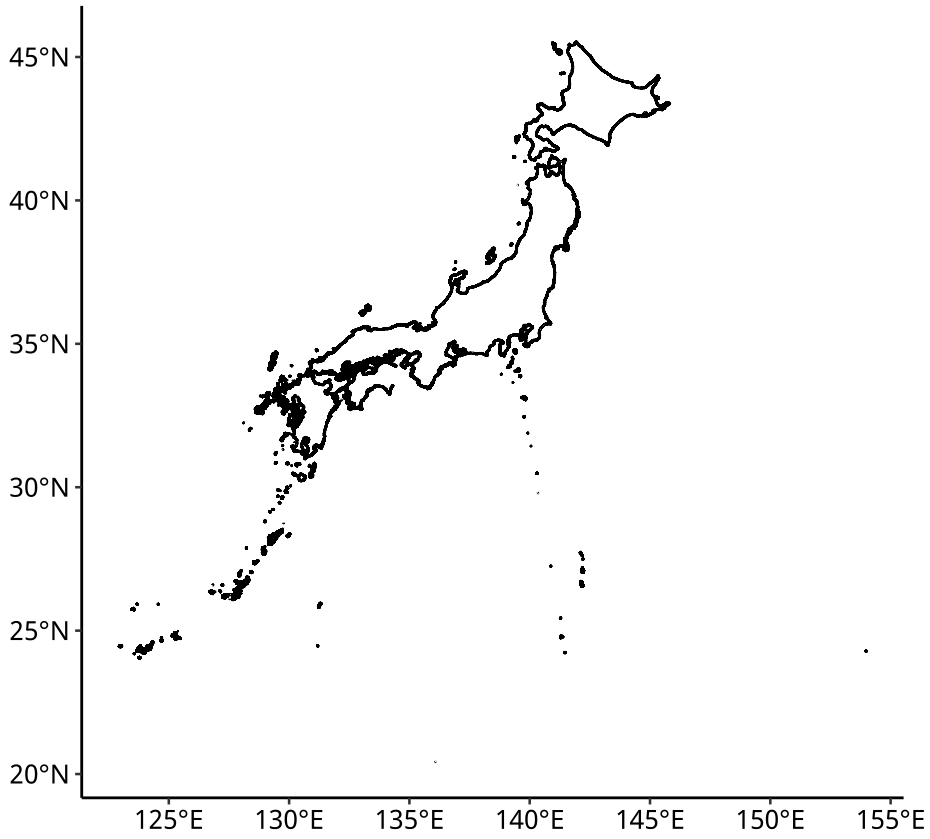
```
      `Units` = c("degrees", "degrees", "meters")) |>
  kbl() |>
  kable_styling(bootstrap_options = c("hover"))
```

RDS st_union() polgyon
unnest simple feature 121158 features

C23

```
mlit = tibble(folder = dir("~/Lab_Data/Japan_map_data/Coastline/", full = TRUE)) |>
  mutate(data = map(folder, read_sf)) |> select(data) |>
  unnest(data) |>
  st_as_sf(crs = st_crs(6668))
```

```
mlit |> ggplot() + geom_sf()
```

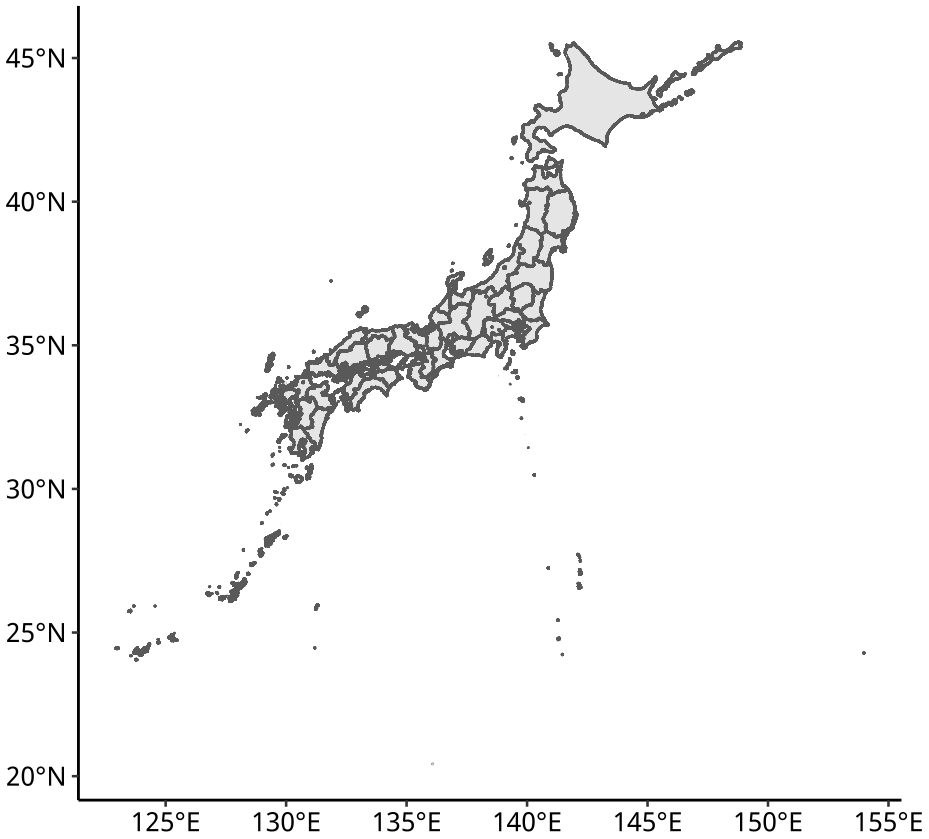


mlit st_union() polygon ~/Lab_Data/Japan_map_data/Jap
RDS

```
# Takes 5.5 hours to complete with 30 cores!
library(furrr)
plan(multisession, workers = 30)
# Group by prefecture
mlit1 = mlit |> group_nest(N03_001) |>
  mutate(data = future_map(data, st_union)) |>
  unnest(data) |> st_as_sf()
mlit1 |> write_rds("~/Lab_Data/Japan_map_data/Japan/todofuken.rds")
```

```
mlit1 = read_rds("~/Lab_Data/Japan_map_data/Japan/todofuken.rds")
```

```
mlit1 |> ggplot() + geom_sf()
```



8.2

GPS tibble

```
zostera = read_csv("~/Lab_Data/matsumuro/Katagami_Bay/longlat_info.csv")
#> Rows: 105 Columns: 6
#> -- Column specification --------------------------------------
#> Delimiter: ","
#> chr  (1): eelgrass
#> dbl  (4): Name, lat, long, coverage(%)
#> dttm (1): datetime
#>
#> i Use `spec()` to retrieve the full column specification for this data.
#> i Specify the column types or set `show_col_types = FALSE` to quiet this message.
zostera |> print(n = 3)
#> # A tibble: 105 x 6
#>    Name   lat  long datetime            eelgrass
#>   <dbl> <dbl> <dbl> <dttm>              <chr>
```

```
#> 1     1  33.0  130. 2021-05-25 09:14:48 absent
#> 2     2  33.0  130. 2021-05-25 09:30:32 absent
#> 3     3  33.0  130. 2021-05-25 09:37:16 present
#> # ... with 102 more rows, and 1 more variable:
#> #   `coverage(%)` <dbl>
```

zostera CRS mlit

```
zostera = zostera |> st_as_sf(coords = c("long", "lat"), crs = st_crs(mlit))
zostera |> print(n = 3)
#> Simple feature collection with 105 features and 4 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: 129.7845 ymin: 32.90032 xmax: 129.806 ymax: 32.95375
#> Geodetic CRS:  JGD2011
#> # A tibble: 105 x 5
#>    Name datetime            eelgrass `coverage(%)`
#> * <dbl> <dttm>              <chr>            <dbl>
#> 1     1 2021-05-25 09:14:48 absent               0
#> 2     2 2021-05-25 09:30:32 absent               0
#> 3     3 2021-05-25 09:37:16 present              5
#> # ... with 102 more rows, and 1 more variable:
#> #   geometry <POINT [°]>
```

8.3

Nagasaki Naoasaki

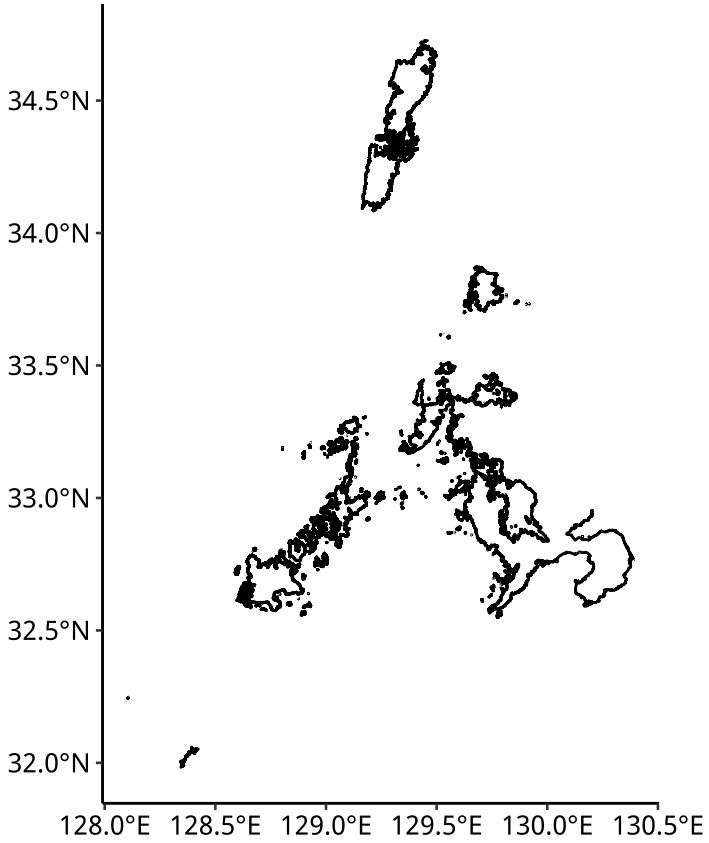
```
toget = " | | | | | | "
kyushu = mlit1 |> filter(str_detect(N03_001, toget))
```

(mlit)

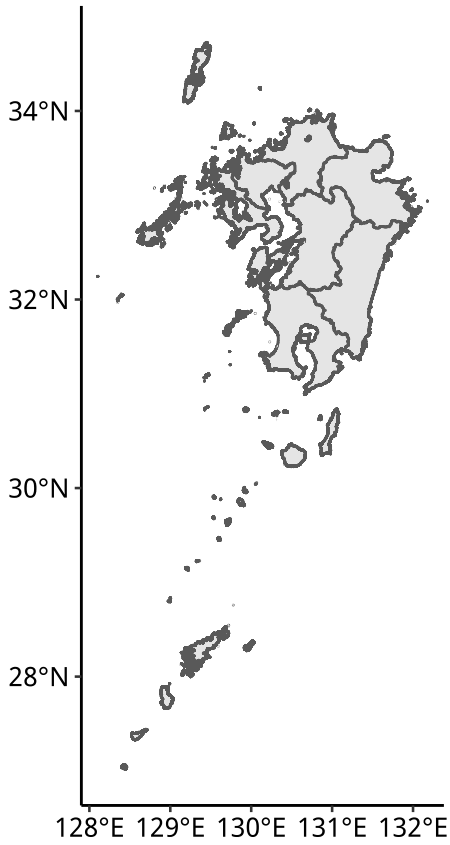
```
admincode = readxl::read_xlsx("~/Lab_Data/Japan_map_data/AdminiBoundary_CD.xlsx", skip
admincode = admincode |> select(code = matches(" "), N03_001 = matches("  *.* "))
codes = admincode |> filter(str_detect(N03_001, " ")) |> pull(code)
```

```
nagasaki = mlit |> filter(str_detect(C23_001, str_c(codes, collapse = "|")))
```

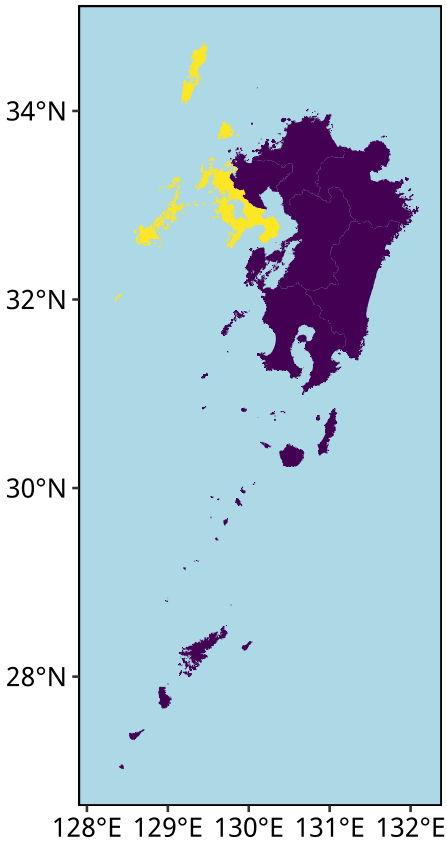
```
ggplot() + geom_sf(data = nagasaki)
```




```
  mlit1
ggplot() + geom_sf(data = kyushu)
```




```
kyushu |>
  mutate(fillme = str_detect(N03_001, " ")) |>
  ggplot() + geom_sf(aes(fill = fillme), color = NA) +
  guides(fill = "none") +
  scale_fill_viridis_d() +
  theme(panel.background = element_rect(fill = "lightblue", color = "black"),
        axis.line = element_blank())
```



kyushu

```
kbbox = kyushu |> st_bbox()
```

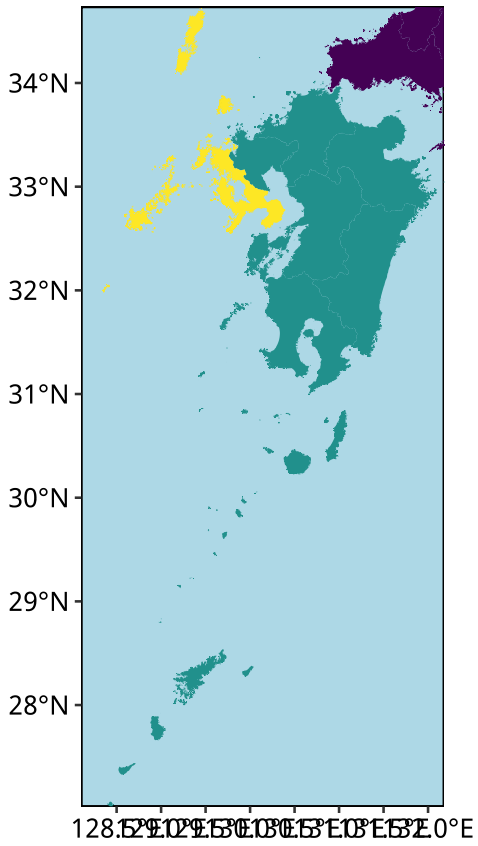
```
toget = " | | | | | | | | | | | | "
kyushu = mlit1 |> filter(str_detect(N03_001, toget))
```

kyushu kbbox

```
kyushu = kyushu |>
  mutate(fillme = case_when(str_detect(N03_001, " ") ~ "Nagasaki",
                            str_detect(N03_001, " | | | | | | ") ~ "Kyushu",
                            TRUE ~ "Honshu")) |>
  st_crop(kbbox)
#> Warning: attribute variables are assumed to be spatially
#> constant throughout all geometries
```

```
ggplot(kyushu) +
  geom_sf(aes(fill = fillme), color = NA) +
```

```
  guides(fill = "none") +
  coord_sf(expand = FALSE) +
  scale_fill_viridis_d() +
  theme(panel.background = element_rect(fill = "lightblue", color = "black"),
        axis.line = element_blank())
```



8.4

Google Map

```
katagami = rbind(rev(c(32.95809069048365, 129.7669185309373)),
                 rev(c(32.89802000729197, 129.82832411747583))) |>
  as_tibble(.name_repair = \(x) c("long", "lat")) |>
  st_as_sf(coords = c("long", "lat"), crs = st_crs(kyushu))

omurabay = rbind(rev(c(33.103196388120104, 129.67183787501082)),
                 rev(c(32.817013859622804, 130.03298144413574))) |>
  as_tibble(.name_repair = \(x) c("long", "lat")) |>
```

```
  st_as_sf(coords = c("long", "lat"), crs = st_crs(kyushu))
```

kyushu

```
omurabay_area = kyushu |> filter(str_detect(N03_001, " ")) |> st_crop(st_bbox(omurabay))
#> Warning: attribute variables are assumed to be spatially
#> constant throughout all geometries
katagami_area = kyushu |> filter(str_detect(N03_001, " ")) |> st_crop(st_bbox(katagami))
#> Warning: attribute variables are assumed to be spatially
#> constant throughout all geometries
```

simple features

```
zostera = zostera |>
  st_as_sf(coords = c("long", "lat"), crs = st_crs(kyushu)) |>
  rename(coverage = matches("cover")) |>
  mutate(rank = cut(coverage,
                    c(-Inf, 1, 10, 40, 70, Inf),
                    labels = c("E", "D", "C", "B", "A"))) |>
  mutate(rank = factor(rank,
                       levels = LETTERS[1:5],
                       labels = LETTERS[1:5]))
```

```
# The main plot of kyushu
pmain = ggplot(kyushu) +
  geom_sf(aes(fill = fillme), color = NA) +
  guides(fill = "none") +
  coord_sf(expand = FALSE) +
  scale_fill_viridis_d() +
  theme(panel.grid = element_blank(),
        panel.background = element_rect(fill = "lightblue", color ="black"),
        panel.border  = element_rect(fill = NA, color ="black"),
        plot.background =  element_rect(fill = NA, color =NA),
        axis.title = element_blank(),
        axis.line = element_blank())
```

tibble tibble long lat

```
# Build plots for Omura Bay and Katagami Bay.
tmp1 = omurabay_area |> st_transform(crs = st_crs(6677)) |> st_bbox()
tmp2 = katagami_area |> st_transform(crs = st_crs(6677)) |> st_bbox()
# tibble for labeling figures. The long and lat are by trial-and-error.
# Need to find a better method.
label1 = tibble(long = tmp1[3] -2500,
                lat = tmp1[2] +1700,
                label = "Omura Bay, Nagasaki, Japan") |>
```

```
  st_as_sf(coords = c("long", "lat"), crs = st_crs(6677), agr = "constant") |>
  st_transform(crs = st_crs(omurabay_area))

label2 = tibble(long = tmp2[1] +800,
                lat = tmp2[4] -150,
                label = "Katagami Bay, Nagasaki, Japan") |>
  st_as_sf(coords = c("long", "lat"), crs = st_crs(6677), agr = "constant") |>
  st_transform(crs = st_crs(omurabay_area))
```

```
pomura = ggplot() +
  geom_sf(fill = "grey50", data = omurabay_area, size = 0) +
  geom_sf_text(aes(label = label),
               data = label1,
               color = "white",
               family = "notosans",
               fontface = "bold",
               vjust = 1, hjust = 1,
               size = 5)  +
  coord_sf(expand = FALSE) +
  annotation_north_arrow(style = north_arrow_minimal(text_family = "notosans",
                                                     text_face = "bold",
                                                     line_width = 2,
                                                     text_size = 20),
                         pad_y = unit(0.3, "npc")) +
  theme(panel.background = element_rect(fill = "lightblue", color ="black"),
        panel.border  = element_rect(fill = NA, color ="black"),
        plot.background =  element_rect(fill = "white", color =NA),
        axis.title = element_blank(),
        axis.line = element_blank(),
        axis.text = element_blank(),
        axis.ticks = element_blank())

pkatagami = ggplot() +
  geom_sf(fill = "grey50", data = katagami_area, size = 0) +
  geom_sf(aes(fill = rank), data = zostera,
          pch = 21, size = 3,
          color = "white", stroke = 1) +
  geom_sf_text(aes(label = label),
               data = label2,
               color = "white",
               family = "notosans",
               fontface = "bold",
               vjust = 1.0, hjust = 0.0,
               size = 5)  +
```

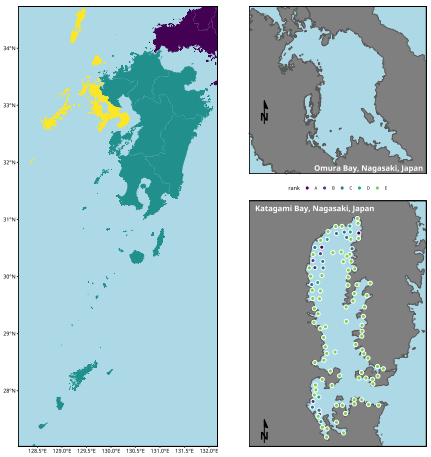
```
  annotation_north_arrow(style = north_arrow_minimal(text_family = "notosans",
                                                     text_face = "bold",
                                                     line_width = 2,
                                                     text_size = 20)) +
  coord_sf(expand = FALSE, crs = st_crs(katagami_area)) +
  scale_fill_viridis_d(end = 0.8) +
  theme(panel.grid = element_blank(),
        panel.background = element_rect(fill = "lightblue", color ="black"),
        panel.border  = element_rect(fill = NA, color ="black"),
        plot.background =  element_rect(fill = "white", color =NA),
        axis.title = element_blank(),
        axis.line = element_blank(),
        axis.text = element_blank(),
        axis.ticks = element_blank())
```

patchwork PDF magick PNG

```
pout = pmain + (pomura / pkatagami)
pdfname = "katagami-map-v1.pdf"
pngname = str_replace(pdfname, "pdf", "png")
ggsave(pdfname, plot= pout, width = 300, height = 300, units = "mm")
#> Warning in st_point_on_surface.sfc(sf::st_zm(x)):
#> st_point_on_surface may not give correct results for
#> longitude/latitude data

#> Warning in st_point_on_surface.sfc(sf::st_zm(x)):
#> st_point_on_surface may not give correct results for
#> longitude/latitude data
image_read_pdf(pdfname, density = 600) |> image_write(pngname)
```

```
knitr::include_graphics(str_c("./", pngname))
```



8.5 Sesssion information

```
sessionInfo()
#> R version 4.1.3 (2022-03-10)
#> Platform: x86_64-pc-linux-gnu (64-bit)
#> Running under: Debian GNU/Linux 11 (bullseye)
#>
#> Matrix products: default
#> BLAS:   /usr/lib/x86_64-linux-gnu/atlas/libblas.so.3.10.3
#> LAPACK: /usr/lib/x86_64-linux-gnu/atlas/liblapack.so.3.10.3
#>
#> locale:
#>  [1] LC_CTYPE=en_US.utf8        LC_NUMERIC=C
#>  [3] LC_TIME=ja_JP.UTF-8        LC_COLLATE=en_US.utf8
#>  [5] LC_MONETARY=ja_JP.UTF-8    LC_MESSAGES=en_US.utf8
#>  [7] LC_PAPER=ja_JP.UTF-8       LC_NAME=C
#>  [9] LC_ADDRESS=C               LC_TELEPHONE=C
#> [11] LC_MEASUREMENT=ja_JP.UTF-8 LC_IDENTIFICATION=C
#>
#> attached base packages:
#> [1] stats     graphics  grDevices utils     datasets
#> [6] methods   base
#>
#> other attached packages:
#>  [1] showtext_0.9-5   showtextdb_3.0   sysfonts_0.8.5
#>  [4] sf_1.0-7         ggspatial_1.1.5  patchwork_1.1.1
#>  [7] magick_2.7.3     kableExtra_1.3.4 ggpubr_0.4.0
#> [10] forcats_0.5.1    stringr_1.4.0    dplyr_1.0.8
#> [13] purrr_0.3.4      readr_2.1.2      tidyr_1.2.0
#> [16] tibble_3.1.6     ggplot2_3.3.5    tidyverse_1.3.1
#>
#> loaded via a namespace (and not attached):
#>  [1] fs_1.5.2           bit64_4.0.5
#>  [3] lubridate_1.8.0    webshot_0.5.2
#>  [5] httr_1.4.2         tools_4.1.3
#>  [7] backports_1.4.1    utf8_1.2.2
#>  [9] R6_2.5.1           KernSmooth_2.23-20
#> [11] DBI_1.1.2          colorspace_2.0-3
#> [13] withr_2.5.0        tidyselect_1.1.2
#> [15] bit_4.0.4          curl_4.3.2
#> [17] compiler_4.1.3     textshaping_0.3.6
#> [19] cli_3.2.0          rvest_1.0.2
#> [21] xml2_1.3.3         bookdown_0.24
#> [23] scales_1.1.1       classInt_0.4-3
#> [25] askpass_1.1        proxy_0.4-26
```

```
#> [27] systemfonts_1.0.4  digest_0.6.29
#> [29] rmarkdown_2.12     svglite_2.1.0
#> [31] pkgconfig_2.0.3    htmltools_0.5.2
#> [33] dbplyr_2.1.1       fastmap_1.1.0
#> [35] highr_0.9          rlang_1.0.2
#> [37] readxl_1.3.1       rstudioapi_0.13
#> [39] generics_0.1.2     farver_2.1.0
#> [41] jsonlite_1.8.0     vroom_1.5.7
#> [43] car_3.0-12         magrittr_2.0.2
#> [45] s2_1.0.7           Rcpp_1.0.8
#> [47] munsell_0.5.0      fansi_1.0.2
#> [49] abind_1.4-5        lifecycle_1.0.1
#> [51] stringi_1.7.6      yaml_2.3.5
#> [53] carData_3.0-5      grid_4.1.3
#> [55] parallel_4.1.3     crayon_1.5.0
#> [57] haven_2.4.3        hms_1.1.1
#> [59] knitr_1.37         pillar_1.7.0
#> [61] ggsignif_0.6.3     wk_0.6.0
#> [63] reprex_2.0.1       glue_1.6.2
#> [65] evaluate_0.15      pdftools_3.1.1
#> [67] qpdf_1.1           modelr_0.1.8
#> [69] vctrs_0.3.8        tzdb_0.2.0
#> [71] cellranger_1.1.0   gtable_0.3.0
#> [73] assertthat_0.2.1   xfun_0.30
#> [75] broom_0.7.12       e1071_1.7-9
#> [77] rstatix_0.7.0      ragg_1.2.2
#> [79] class_7.3-20       viridisLite_0.4.0
#> [81] units_0.8-0        ellipsis_0.3.2
```

Bibliography

Xie, Y. (2015). *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition. ISBN 978-1498716963.

Xie, Y. (2021). *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.24.