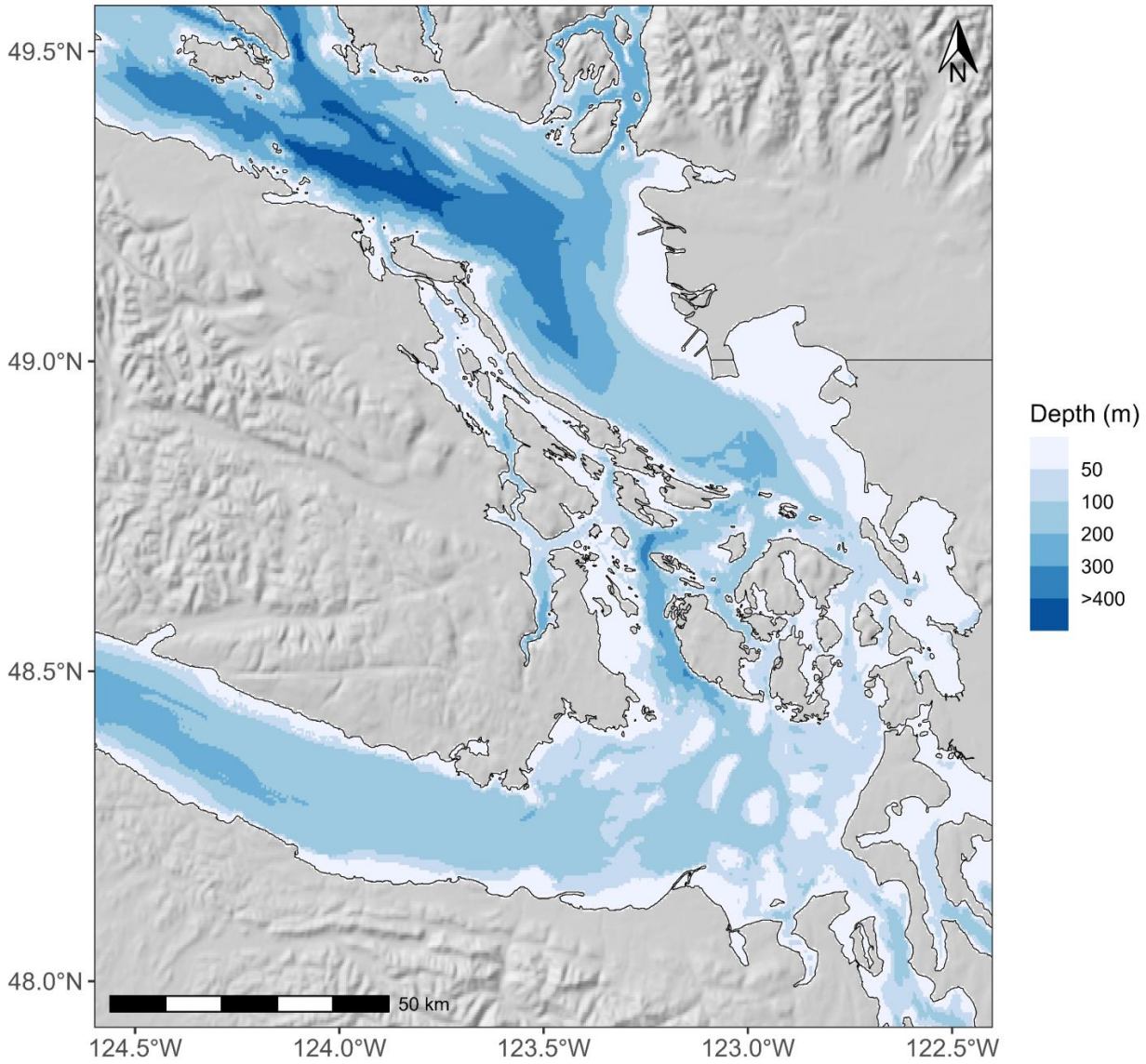
Making maps in R: applications to ecology and evolution

Wesley Greentree
Canadian Society for Ecology and Evolution
May 26, 2024

wgreentree@outlook.com



Juanes Lab

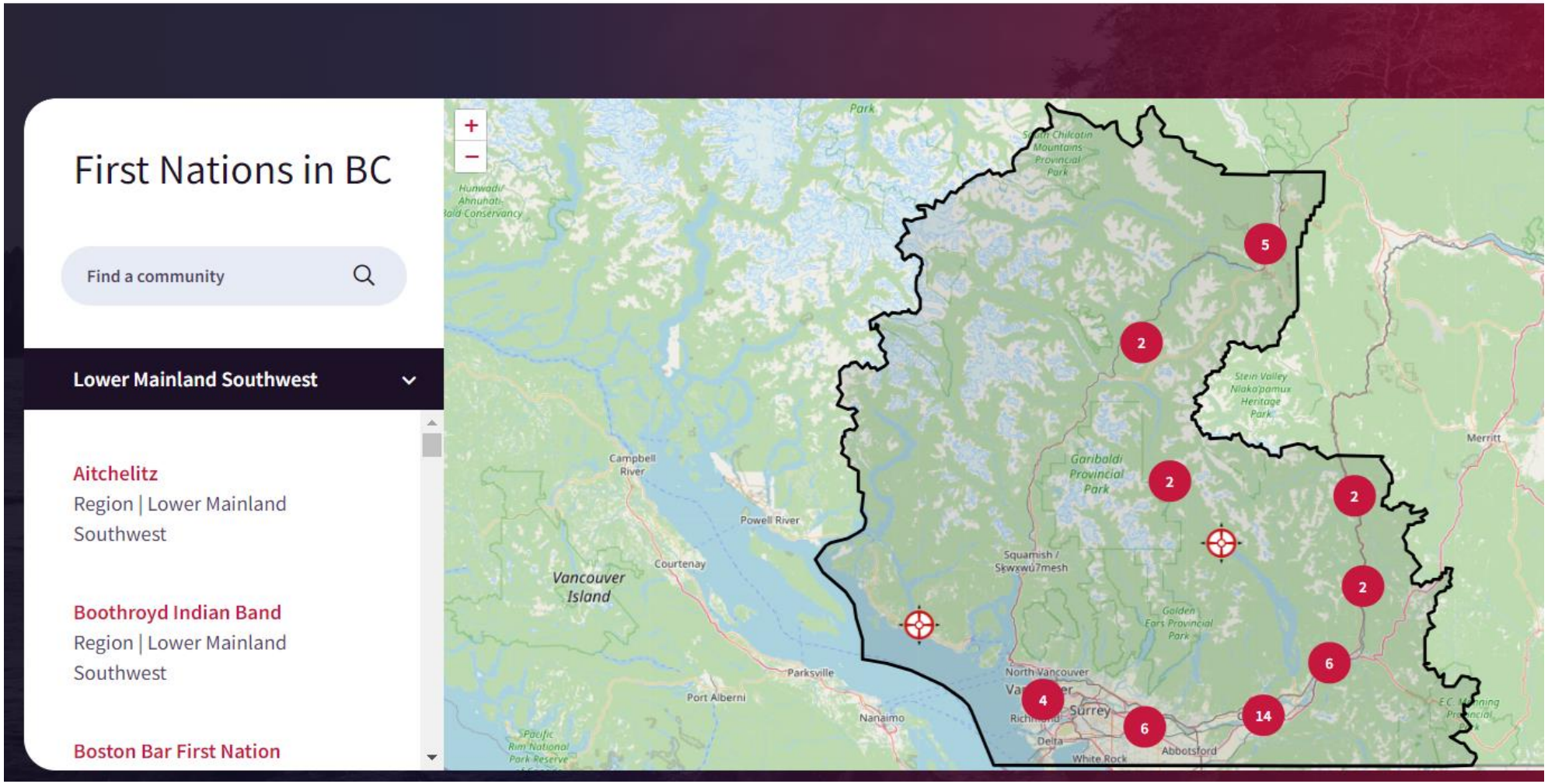
canadian institute of ecology and evolution
institut canadien d'écologie et d'évolution

Territory acknowledgements

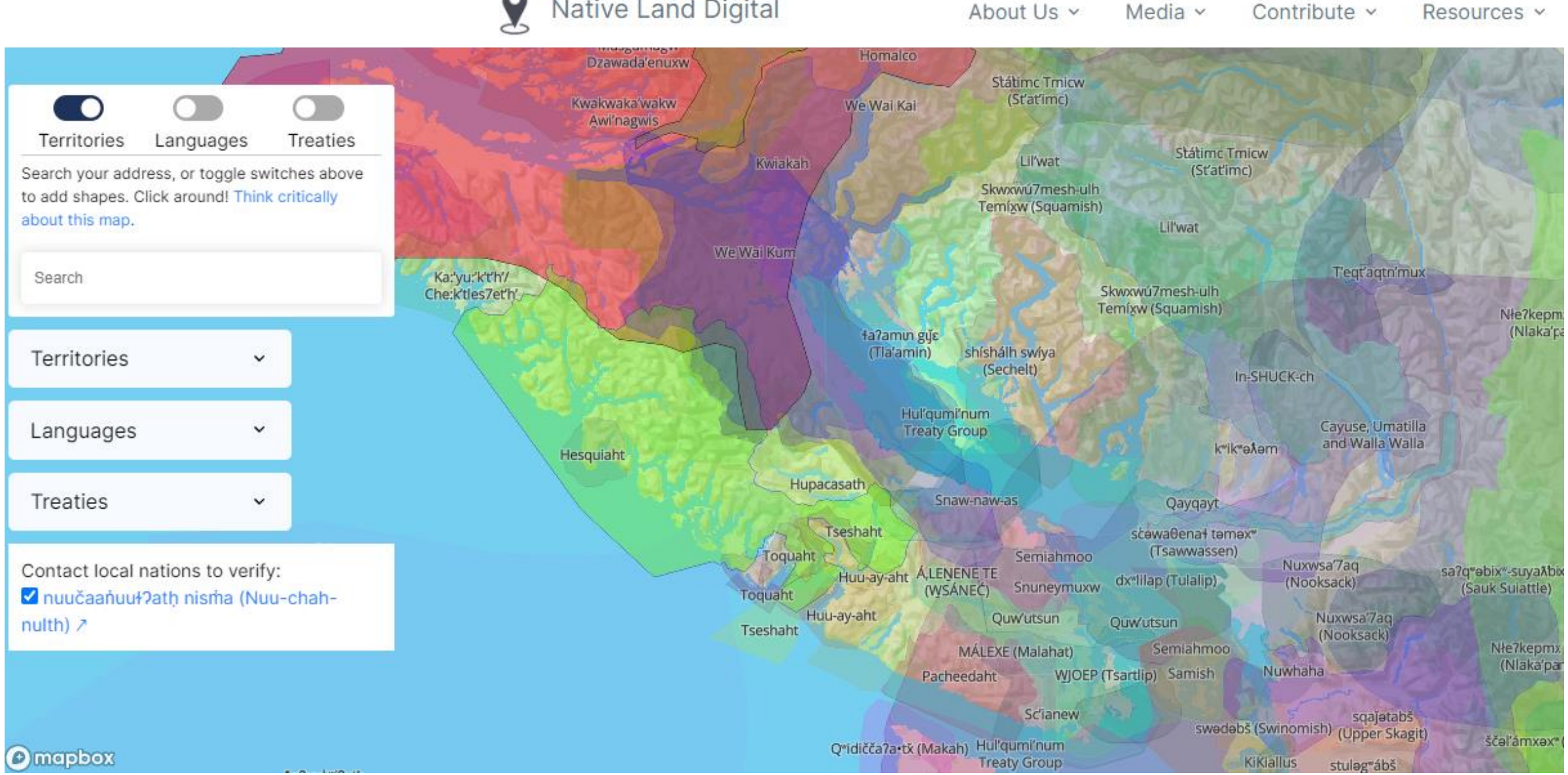
- **UBC** stands on the unceded territory of the Coast Salish Peoples, including the territories of the xwməθkwəy̓əm (Musqueam), Skwxwú7mesh (Squamish), Stó:lō and Sə̓lílwətaʔ/Selilwitulh (Tsleil- Waututh) Nations
- **UVic:** I acknowledge and respect the lək̓ʷəŋən peoples on whose traditional territory the university stands and the Songhees, Esquimalt and W̱SÁNEĆ peoples whose historical relationships with the land continue to this day.

British Columbia Assembly of First Nations

Home | About BCAFN | Priority Areas | News | Publications | Events & Meetings



https://www.bcafn.ca/first-nations-bc/interactive-map?region=10



https://native-land.ca/

Wesley Greentree (he/him)
PhD student, UVic

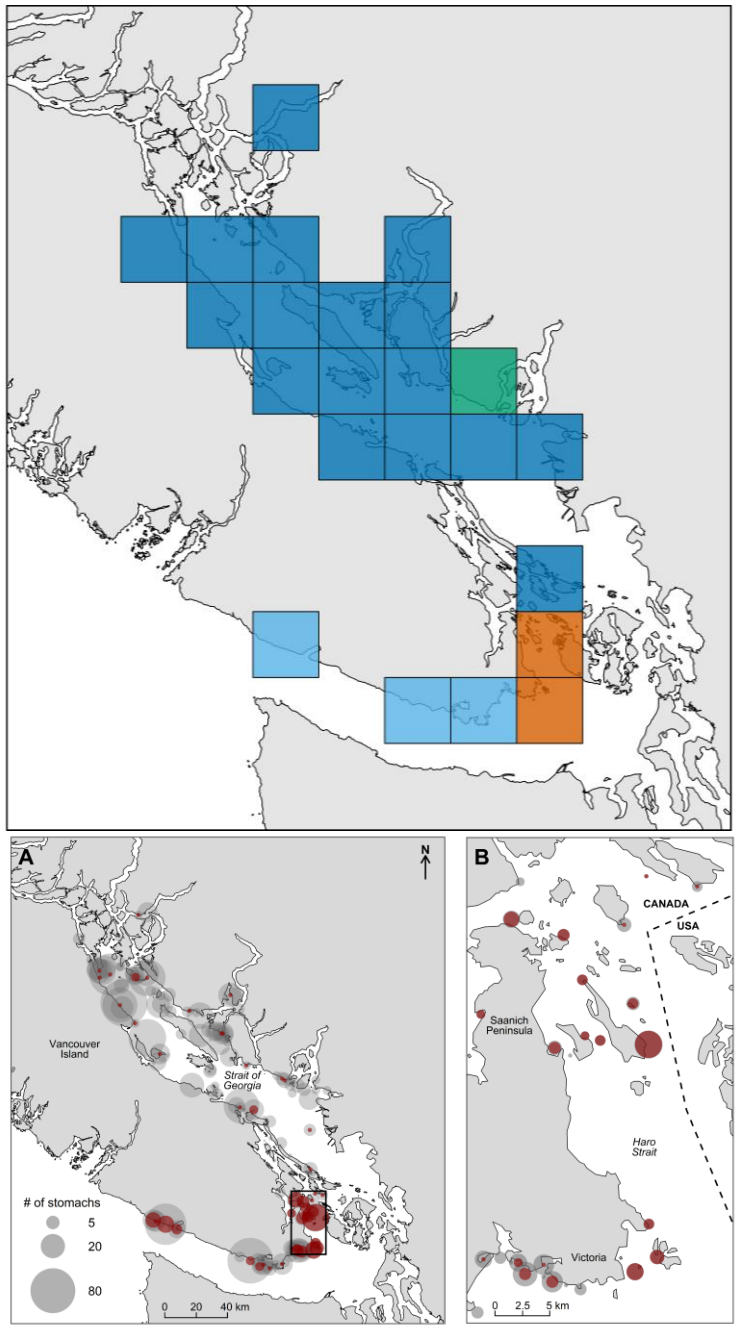
Juanes Lab

PACIFIC SALMON FOUNDATION

Wesley Greentree (he/him)
PhD student, UVic

Juanes Lab

PACIFIC SALMON FOUNDATION



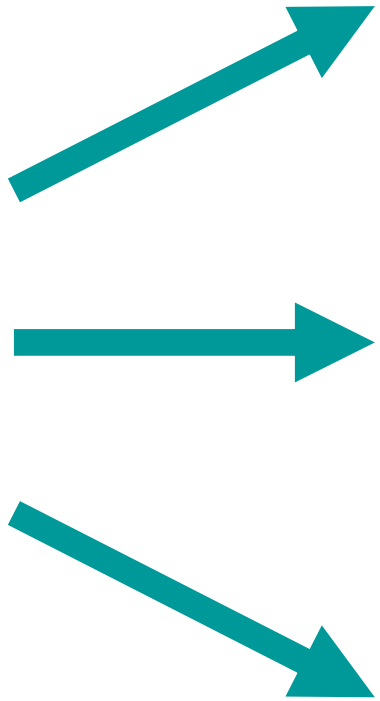
Regional food webs

Salmon migrations

Date: 2023-01-15

Communicate your science through maps

Communicate your science through maps



All the relevant information, but not too much

Clean, beautiful maps

Easy for you!

Workshop plan

Inclusive for beginner R users, with extra info for experienced coders

Topics:

- Introduction to spatial data
- Introduction to `ggplot()`
- Simple study area maps
- Adding additional layers
- Advanced/fun topics

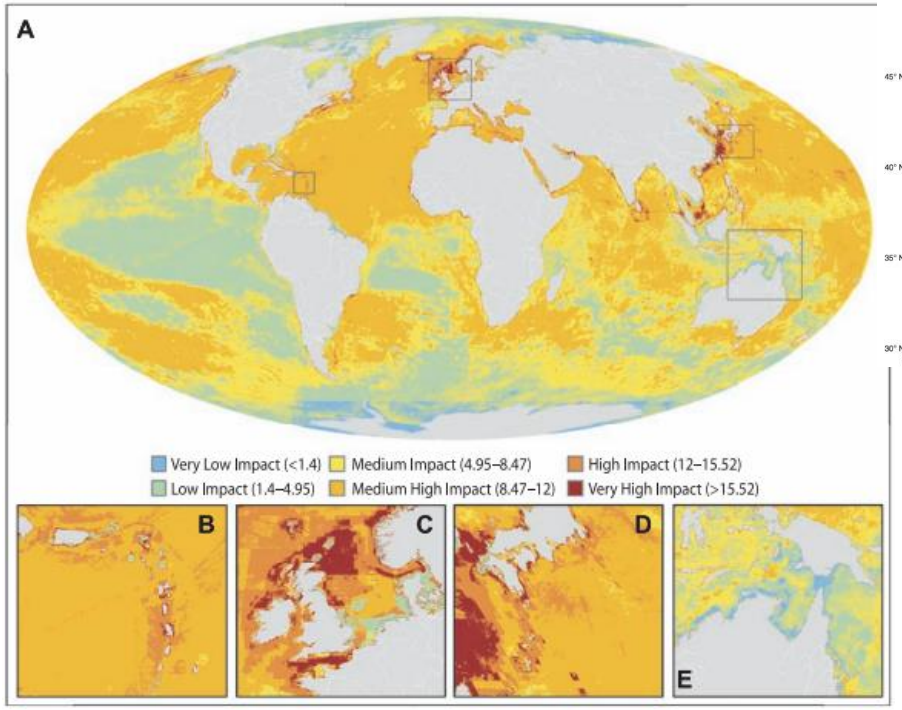
Please interrupt with questions (or raise hand)
Short breaks for discussion/practice

Workshop plan

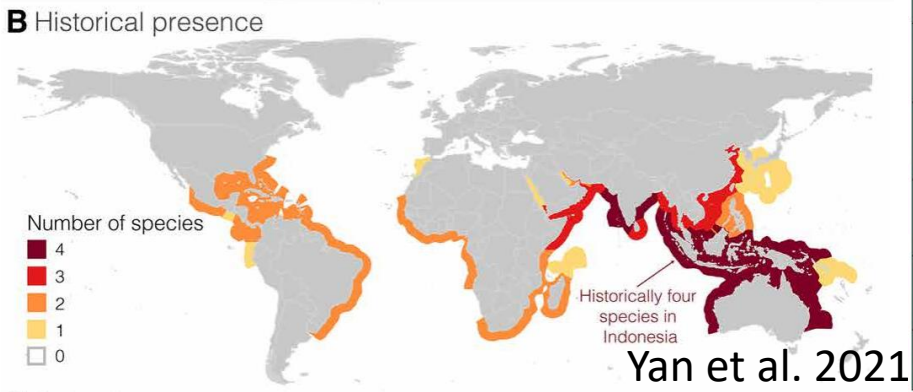
Download code and data from:
https://github.com/wesleygreentree/CSEE2024-R-maps

Spatial data are common in ecology

Global

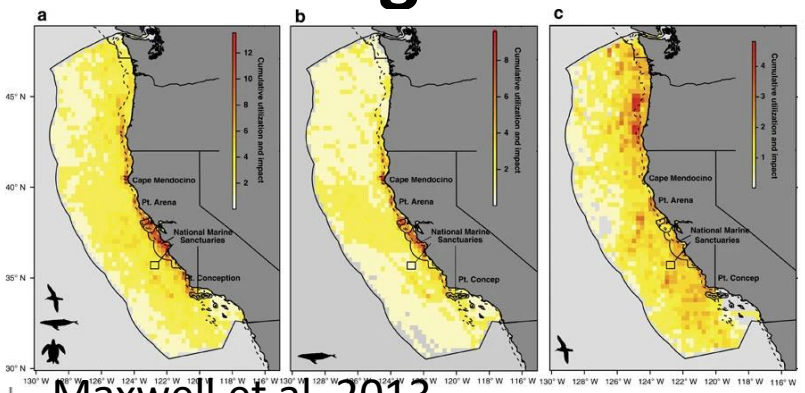


Halpern et al. 2008

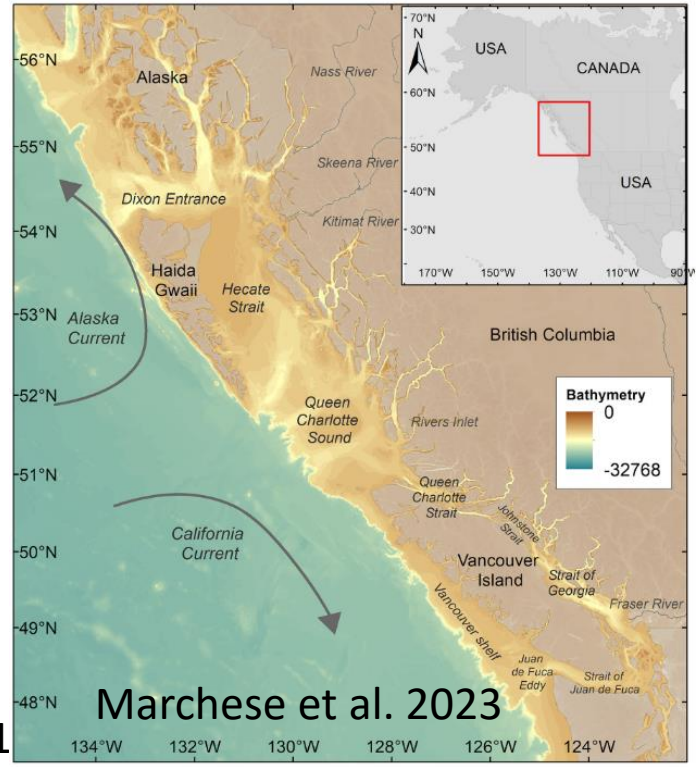


Yan et al. 2021

Regional

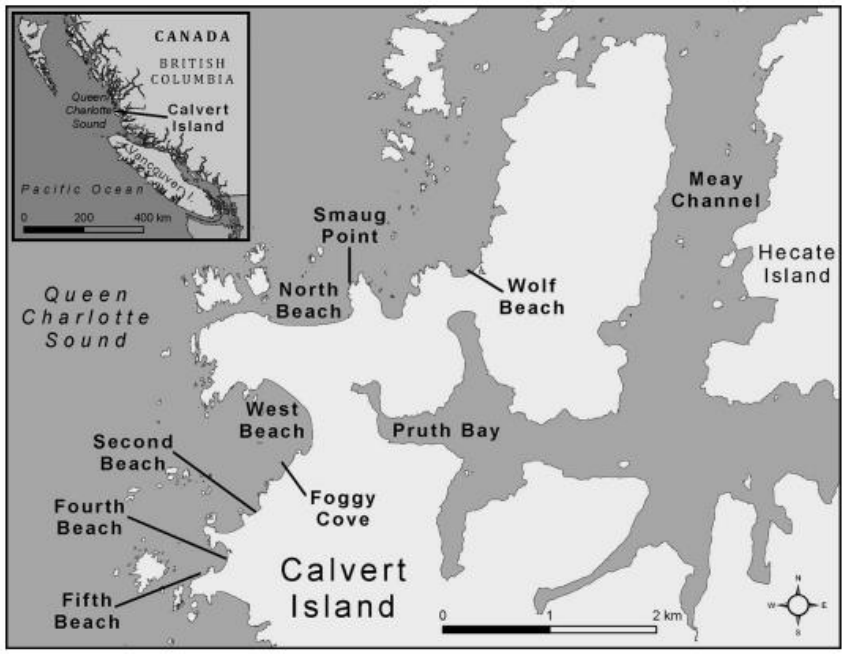


Maxwell et al. 2013

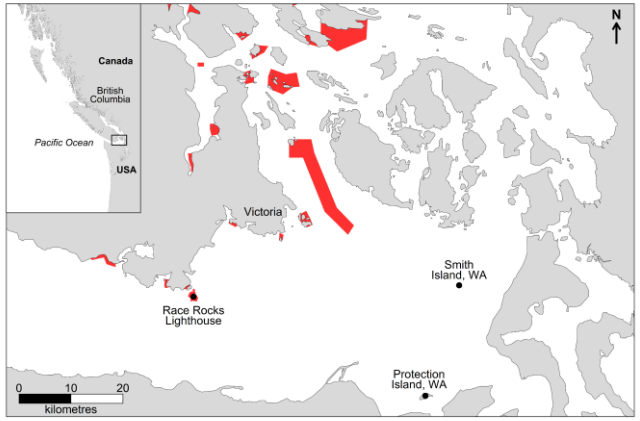


Marchese et al. 2023

Local



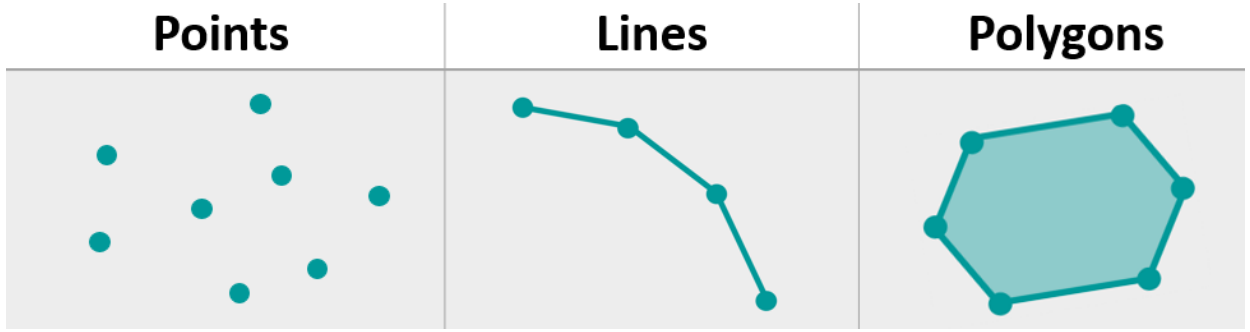
Lindstrom et al. 2021



Types of spatial data

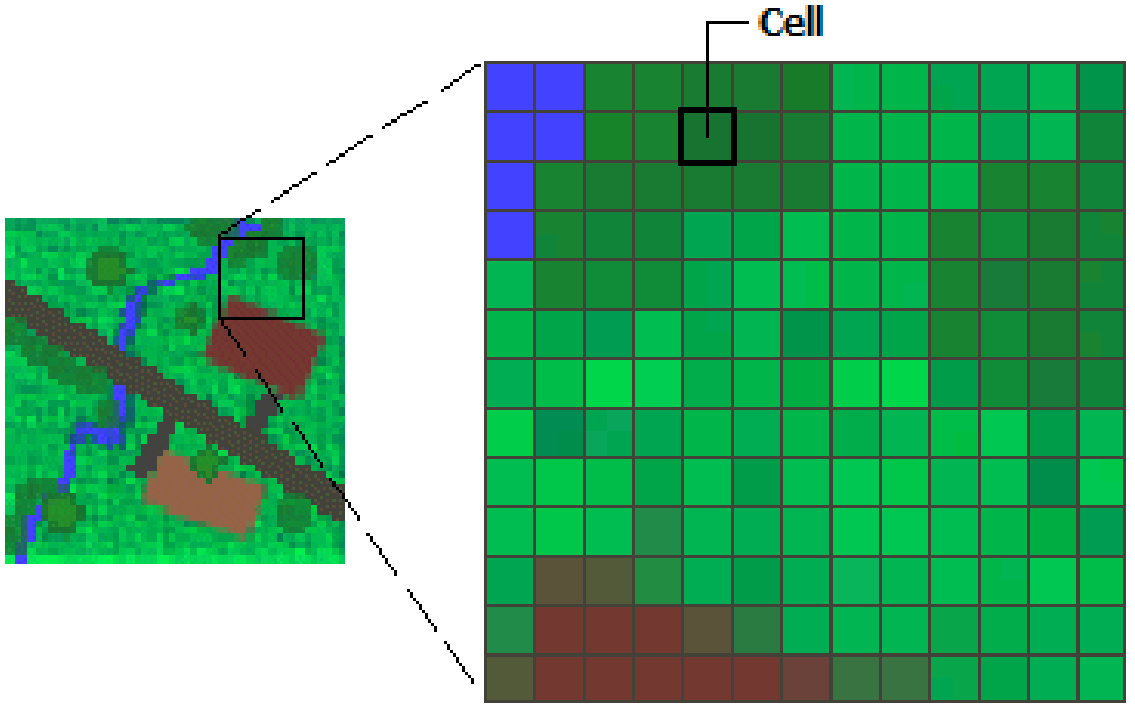
Vectors

coordinates (can be joined together)



Rasters

pixels

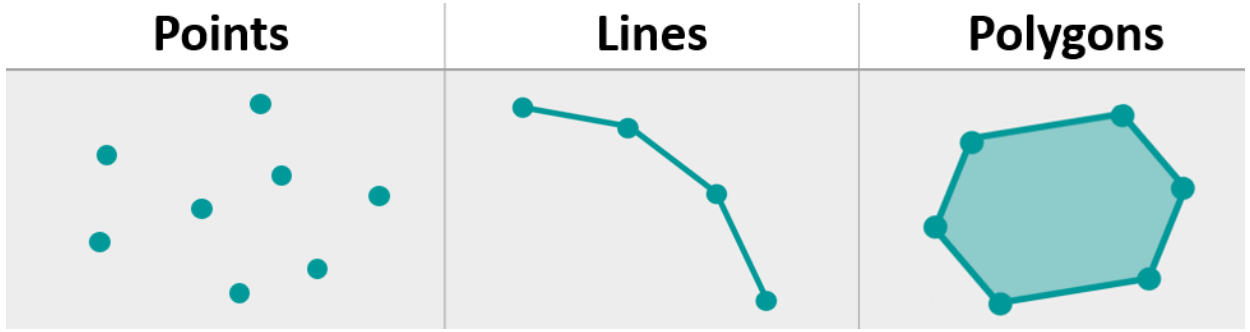


ArcGIS

Types of spatial data

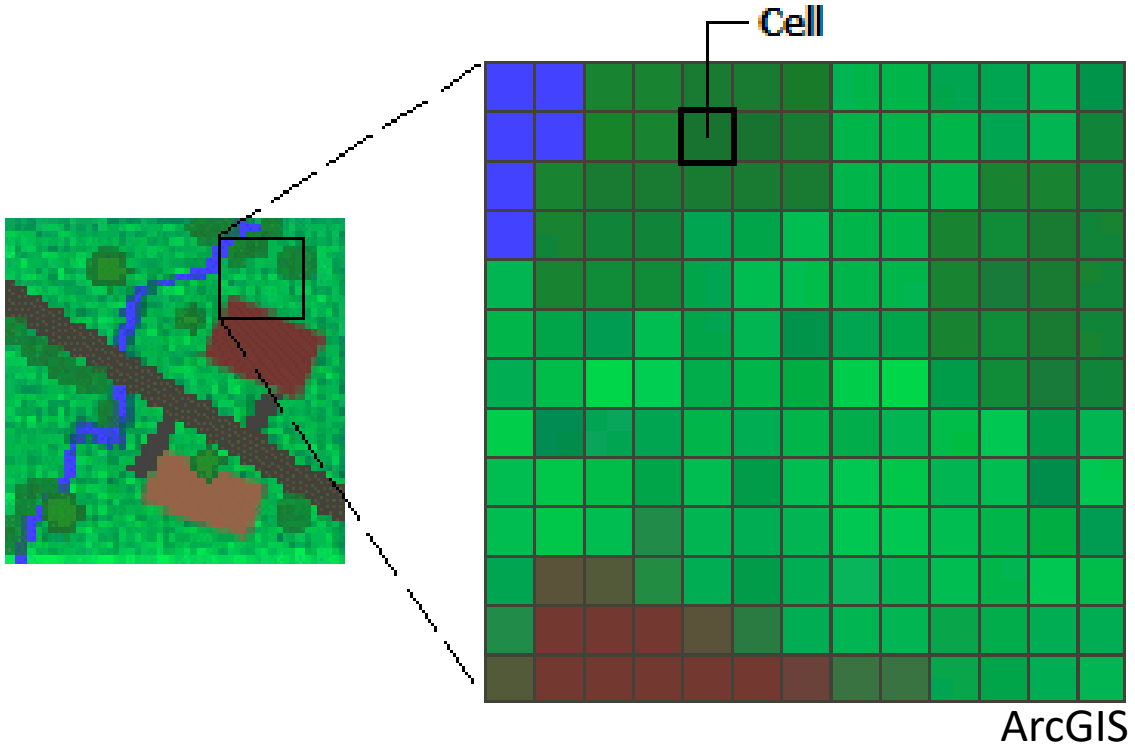
Vectors

coordinates (can be joined together)



Rasters

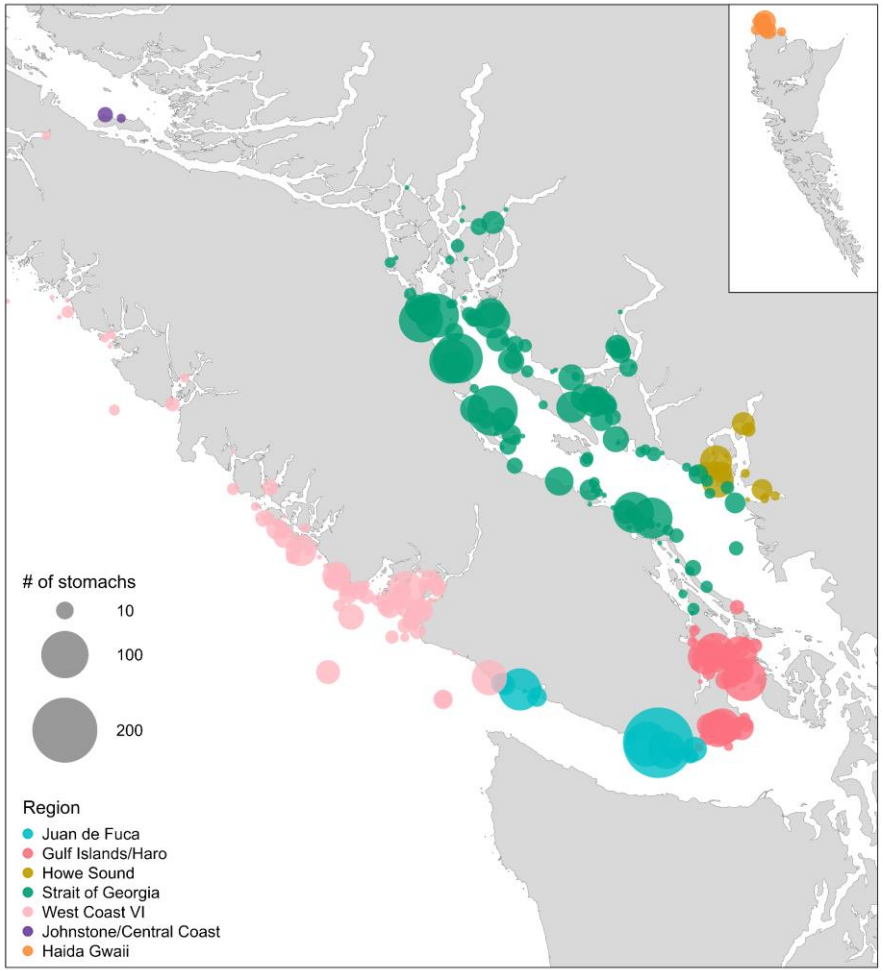
pixels



Types of spatial data

Vectors

polygons, points, lines

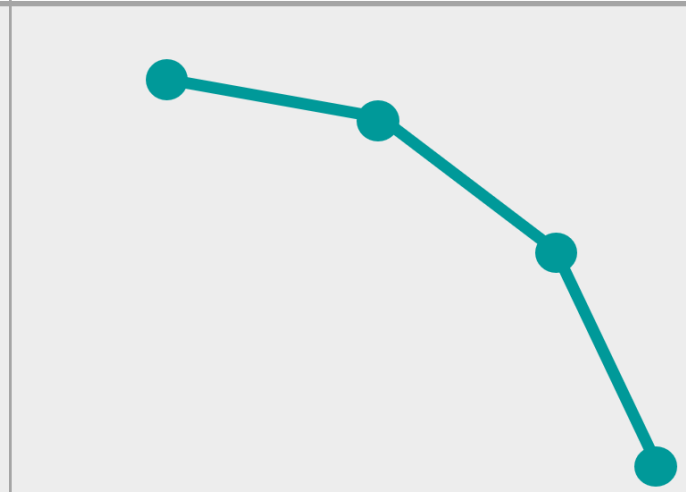
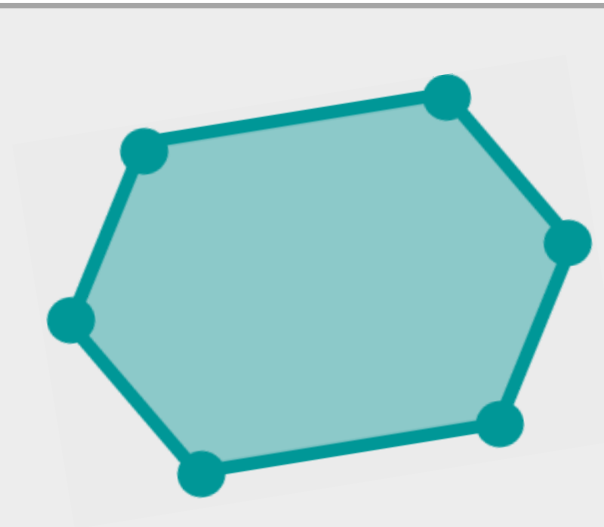


Rasters

pixels

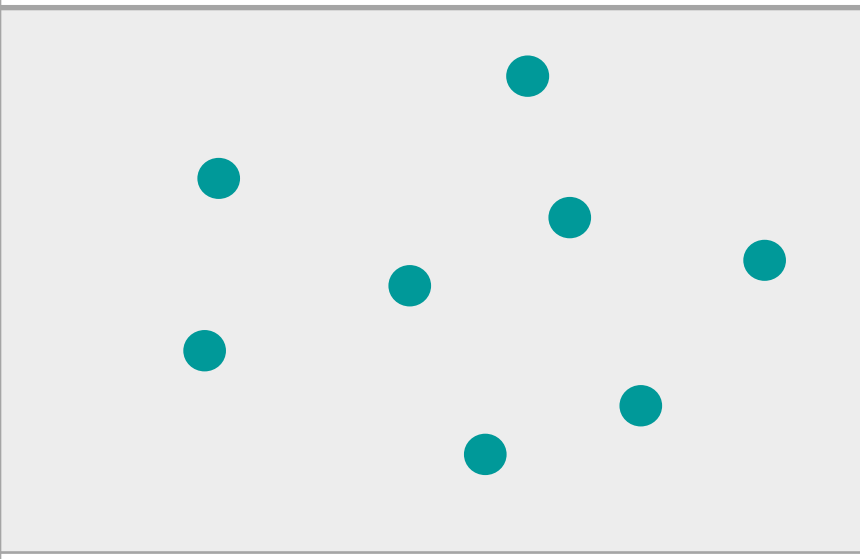
Types of vector data

https://datacarpentry.org/organization-geospatial/02-intro-vector-data/

Points	Lines	Polygons
		
individual x, y positions	lines connect 2+ points	3+ points that connect and close
examples: sampling locations species presence	migration paths roads	coastlines protected areas

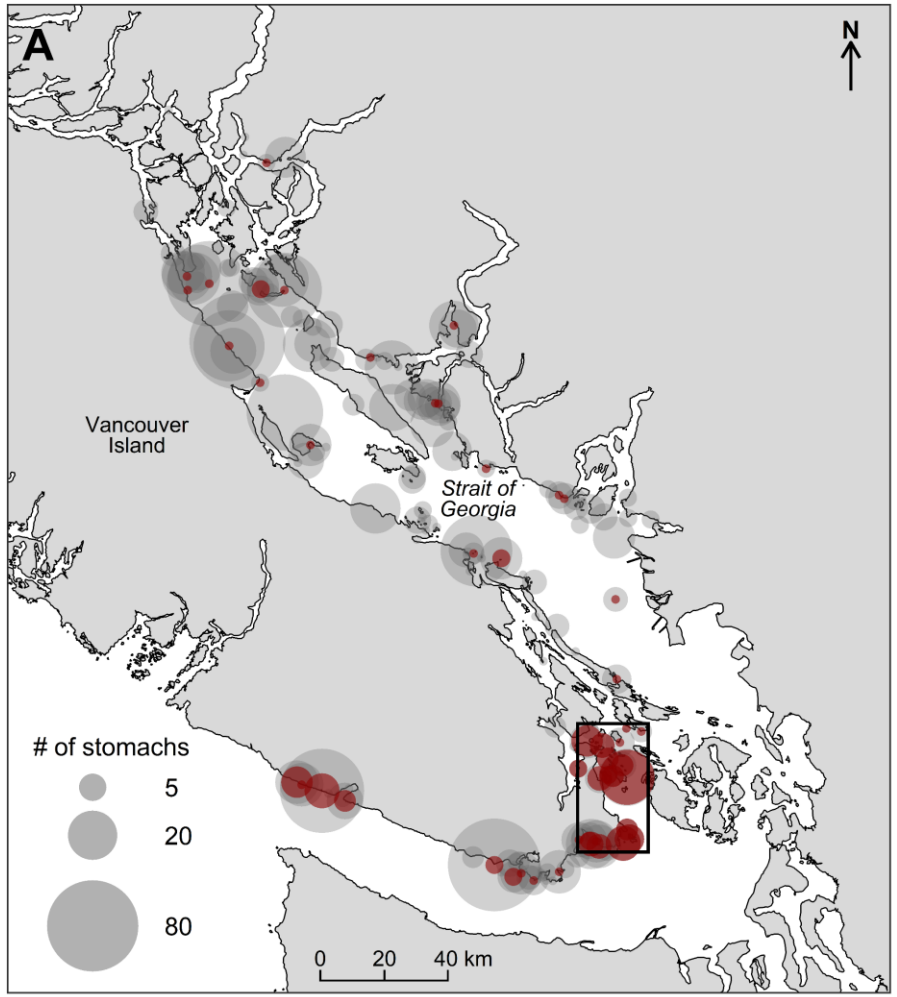
Types of vector data

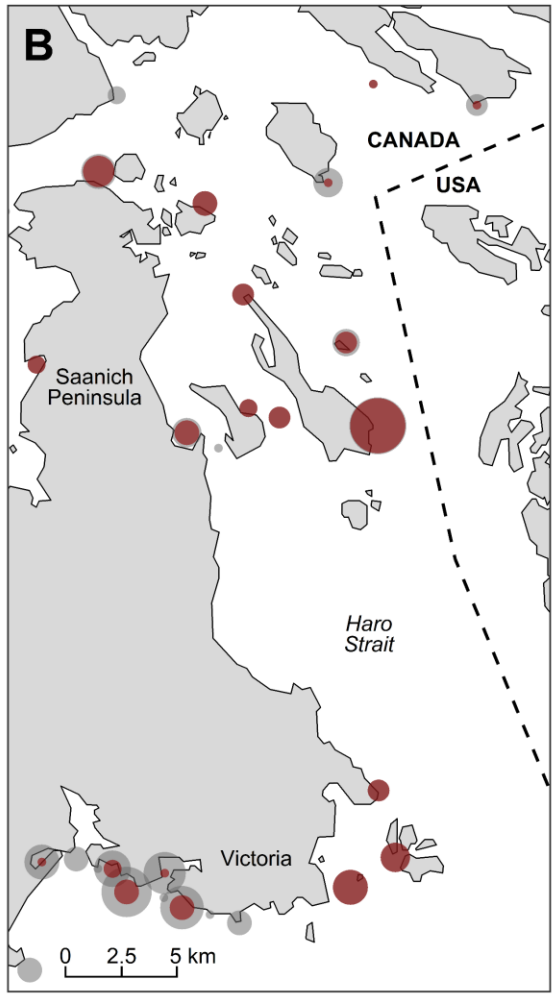
Points



individual x, y positions

examples:
sampling locations
species presence



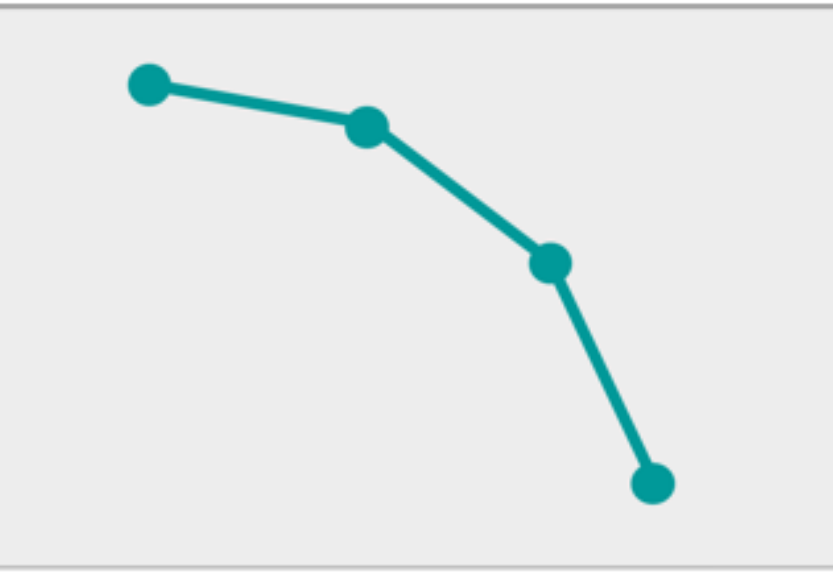


Point size corresponds to number of salmon stomachs containing sand lance

Robinson et al. 2023

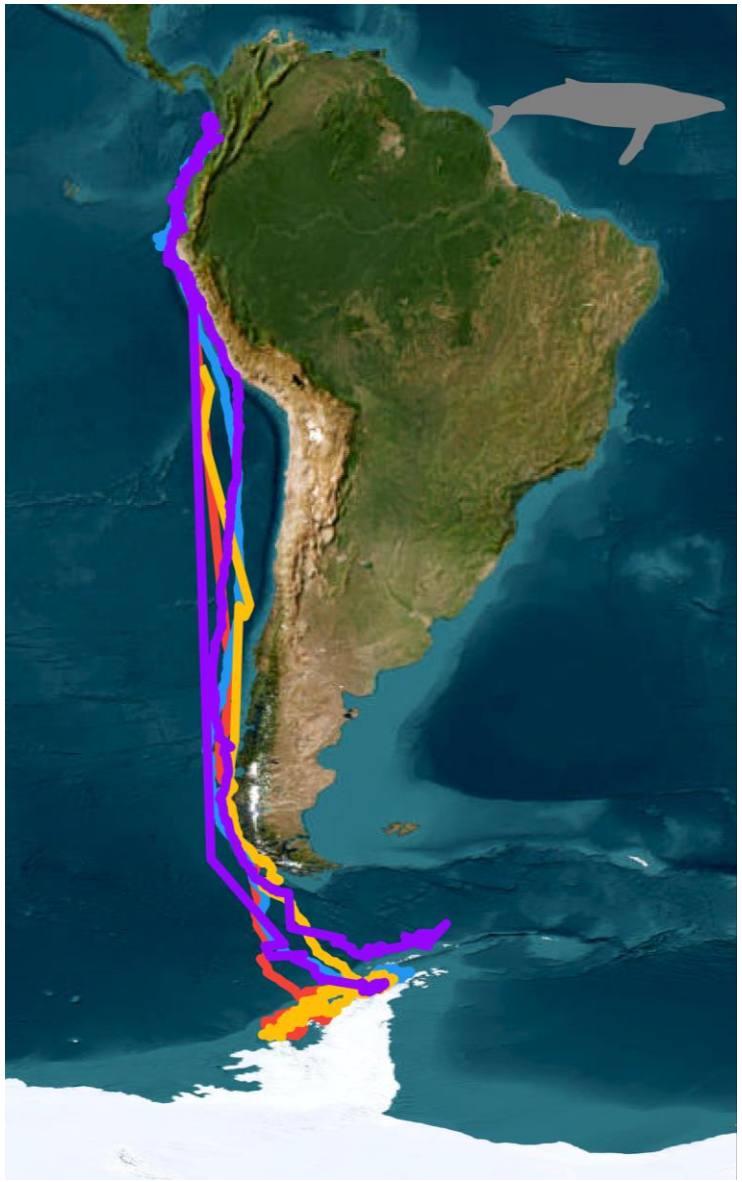
Types of vector data

Lines



lines connect 2+ points

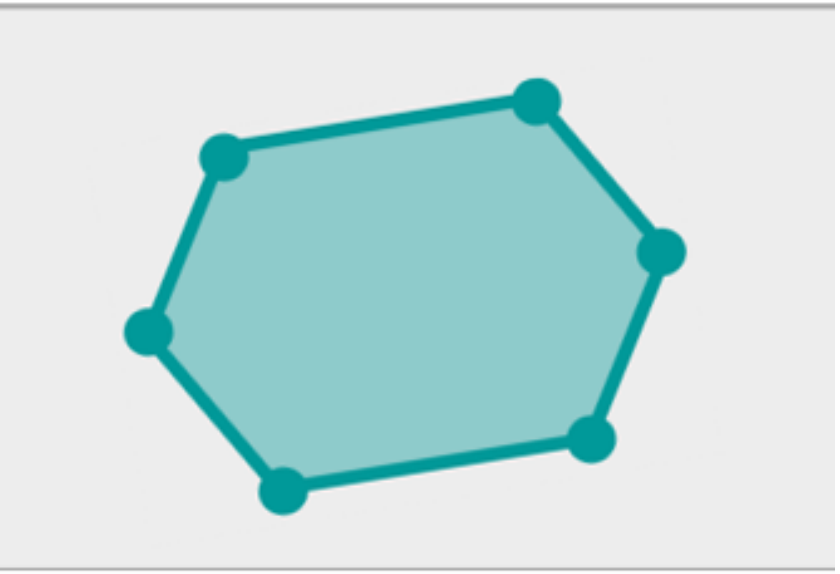
migration paths
roads



Humpback whale migration tracks along South America

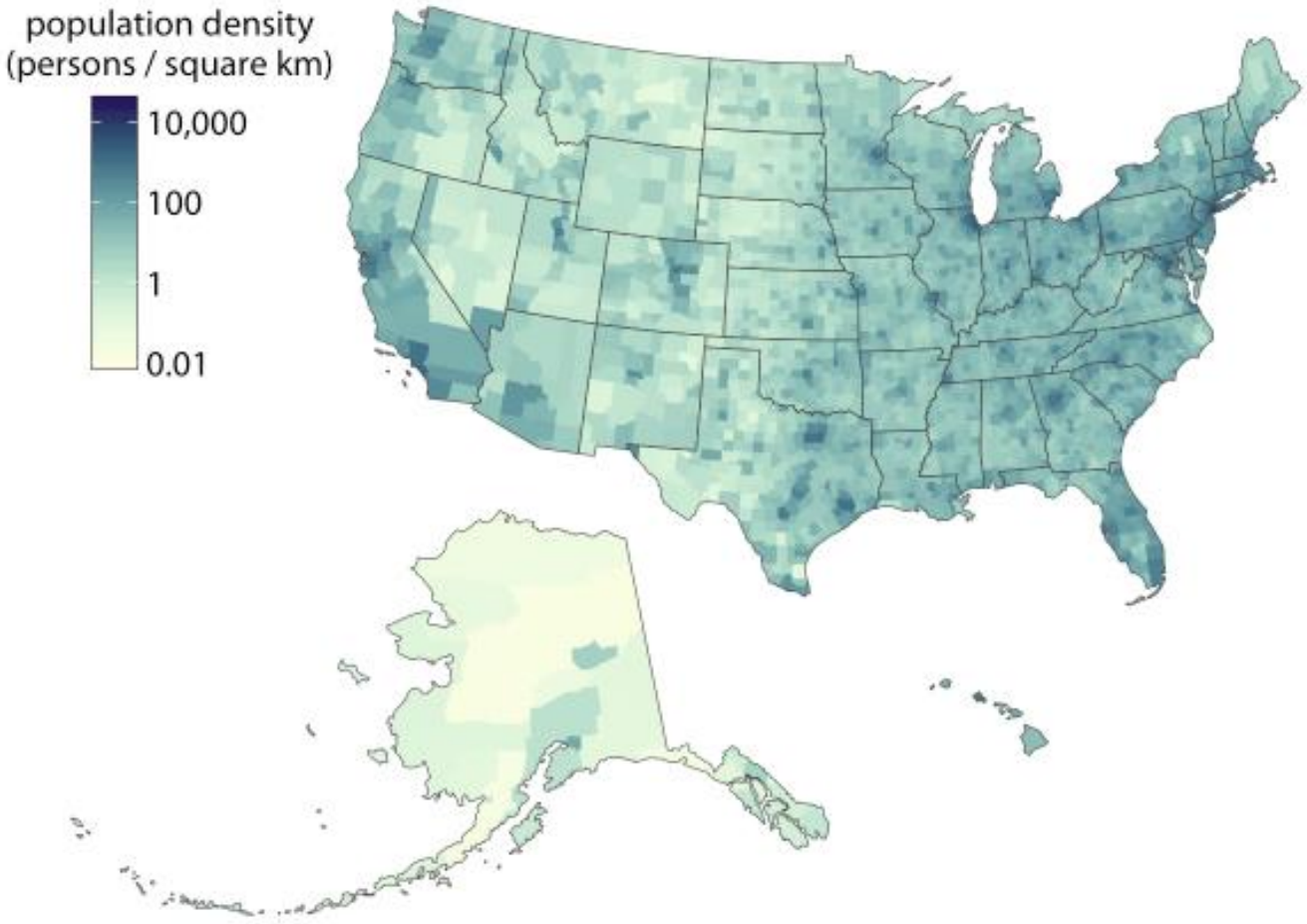
Types of vector data

Polygons



3+ points that connect and close

coastlines
protected areas



Wilke 2019

Why use R for maps?

Challenges with GIS software:

- Most ecologists lack GIS training
- Outside of R data/stats workflow

ArcGIS

QGIS

Why use R for maps?

Challenges with GIS software:

- Most ecologists lack GIS training
- Outside of R data/stats workflow

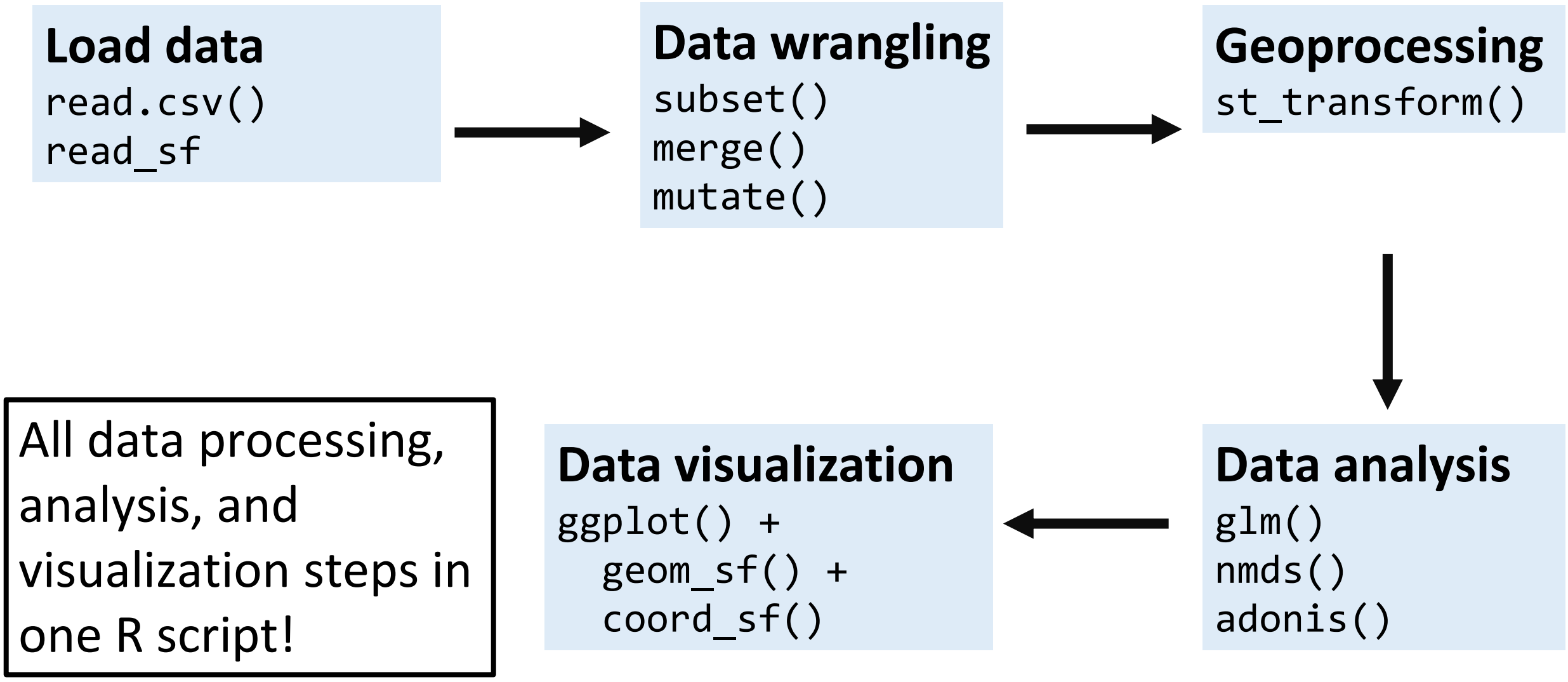
ArcGIS

QGIS

Making maps in R:

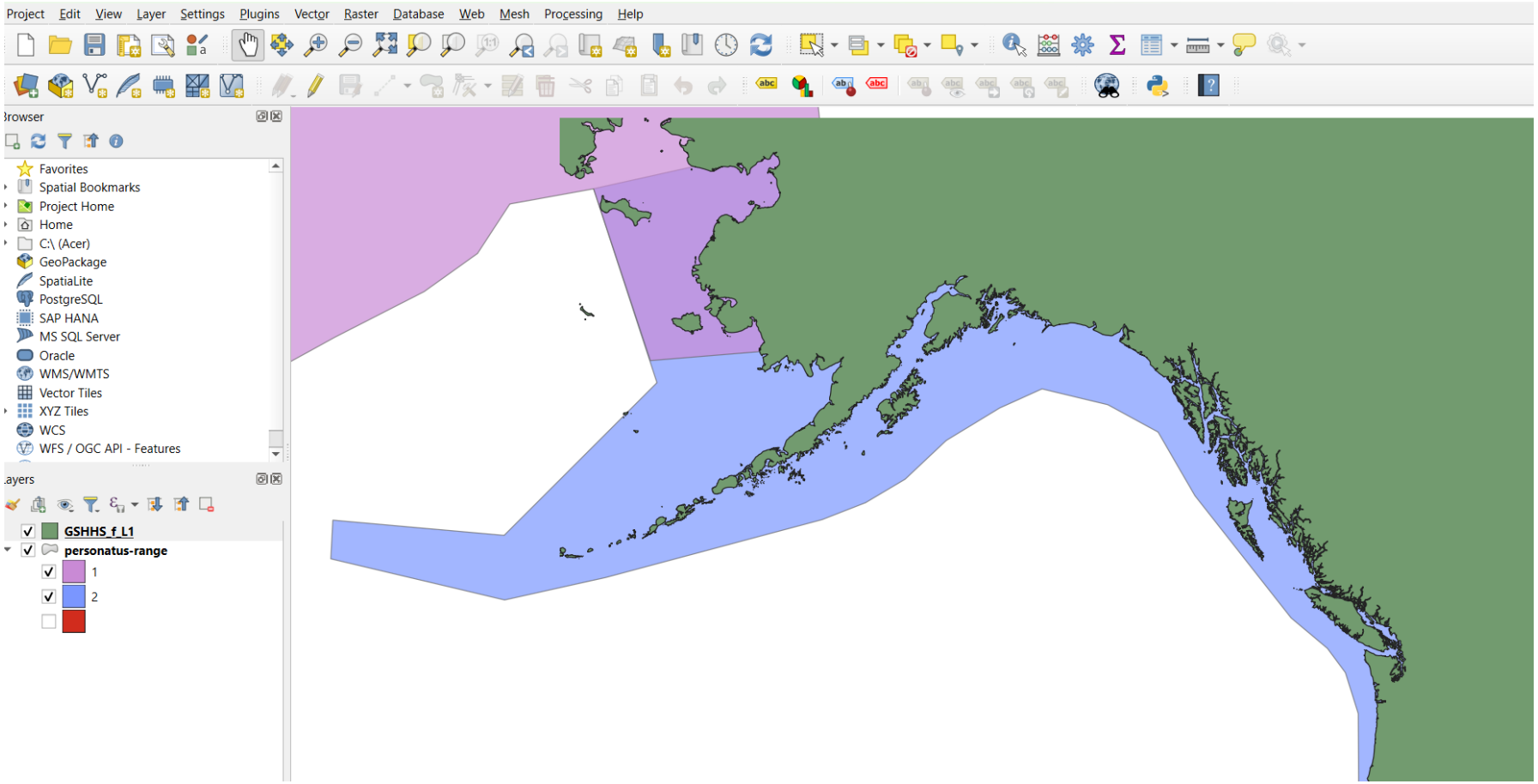
- **Fits in data processing/analysis scripts**
- Reproducibility (workflow saved in script)
- Nice basemaps are built in to R packages
- Faceted maps (many maps in one plot)

Example data analysis workflow



When do I use GIS software?

- Hand-drawing new polygons (e.g., species ranges)
- Producing files for a GPS device



Functions in R

- R uses functions

  `plot()` makes graphs

- Some functions are automatically loaded in R **("base")**

- R software developers build **packages**, which include their functions

```
base::plot()
```

package function

Functions in R

- R uses functions

  `plot()` makes graphs

- Some functions are automatically loaded in R **("base")**

- R software developers build **packages**, which include their functions

  `base::plot()` `ggplot2::ggplot()`

  Both `plot()` and `ggplot()` make plots

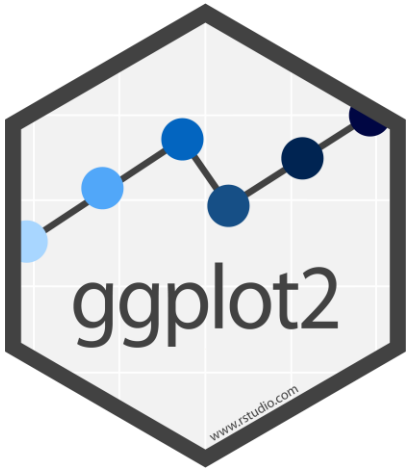
  `ggplot()` makes plotting easier

Functions in R

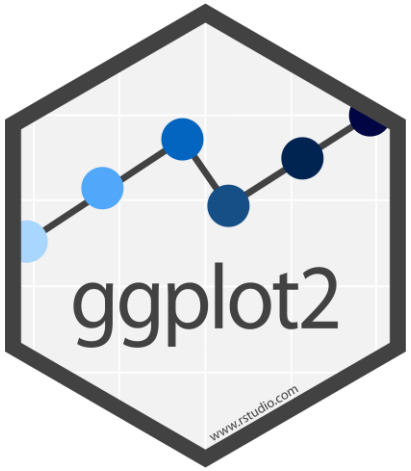
- R uses functions
  `plot()` makes graphs
- Some functions are automatically loaded in R **(“base”)**
- R software developers build **packages**, which include their functions

```
install.packages(“ggplot2”) # first time installing package

library(ggplot2) # load package that is already installed

ggplot2::ggplot() # use a function without loading entire package
```

`ggplot()` “grammar of graphics” in R



`ggplot()` "grammar of graphics" in R



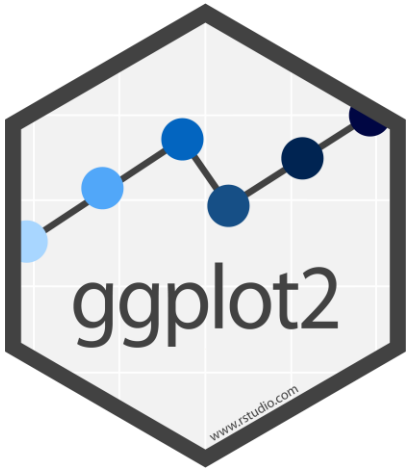
Create figures with multiple layers

Different types of "**geoms**" to make different plots:

- `geom_point()` scatterplots, spatial points
- `geom_line()`, `geom_path()` line graphs, spatial lines
- `geom_sf()` shapefiles (spatial polygons)
- `geom_label()` print text labels on figure

```
ggplot() +
  geom_point(data = df,
             aes(x = X, y = Y, color = group))
```

`ggplot()` "grammar of graphics" in R



Create figures with multiple layers

Different types of "**geoms**" to make different plots:

- `geom_point()` scatterplots, spatial points
- `geom_line()`, `geom_path()` line graphs, spatial lines
- `geom_sf()` shapefiles (spatial polygons)
- `geom_label()` print text labels on figure

data

aesthetics

```
ggplot() +
  geom_point(data = df,
             aes(x = X, y = Y, color = group))
```

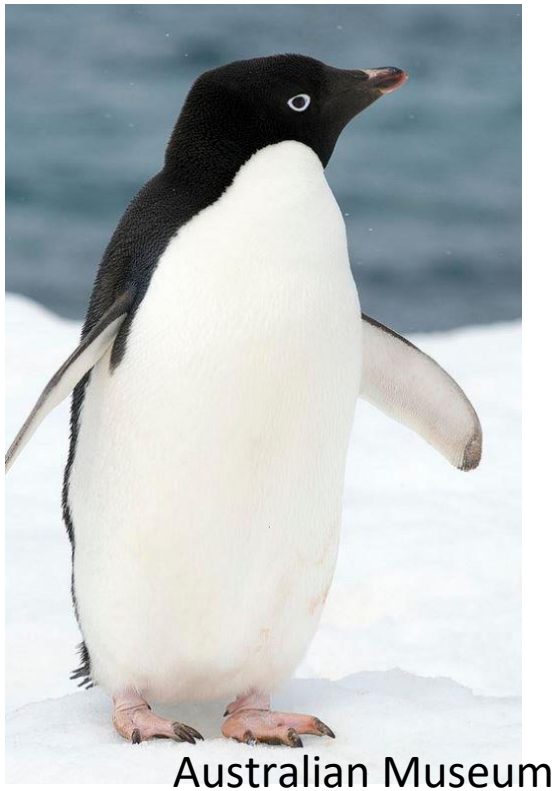
`ggplot()` example

- `palmerpenguins` dataset

- morphological data for 3 penguin species
- example: bill depth vs bill length

https://allisonhorst.github.io/palmerpenguins/

eBird

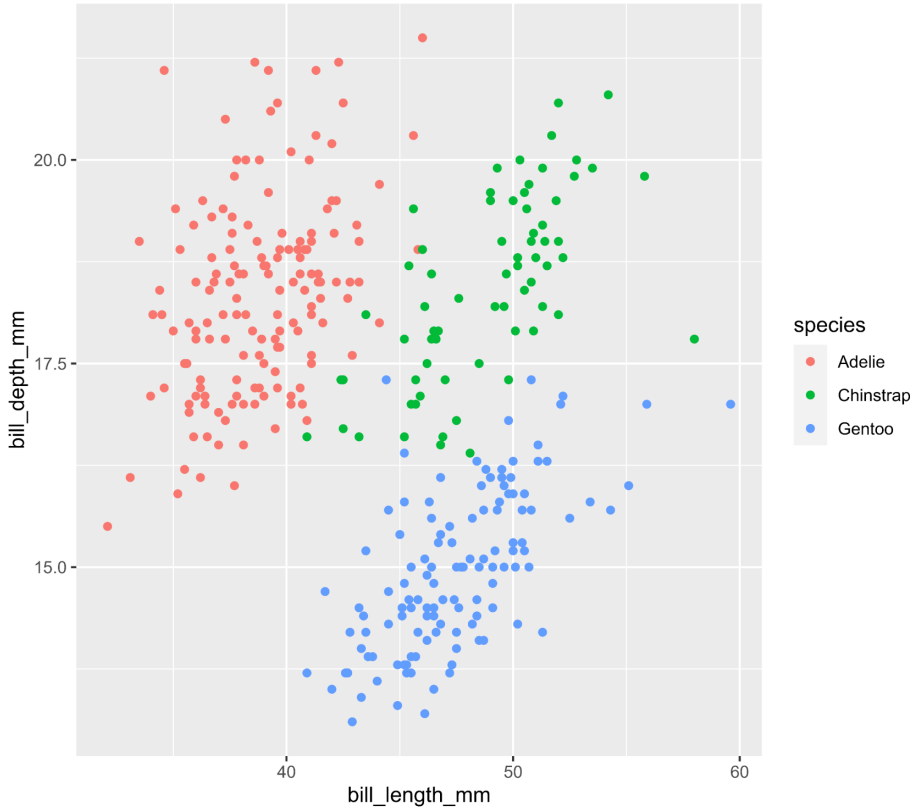


Australian Museum

`ggplot()` example

- `palmerpenguins` dataset
- scripts/ggplot-example.R

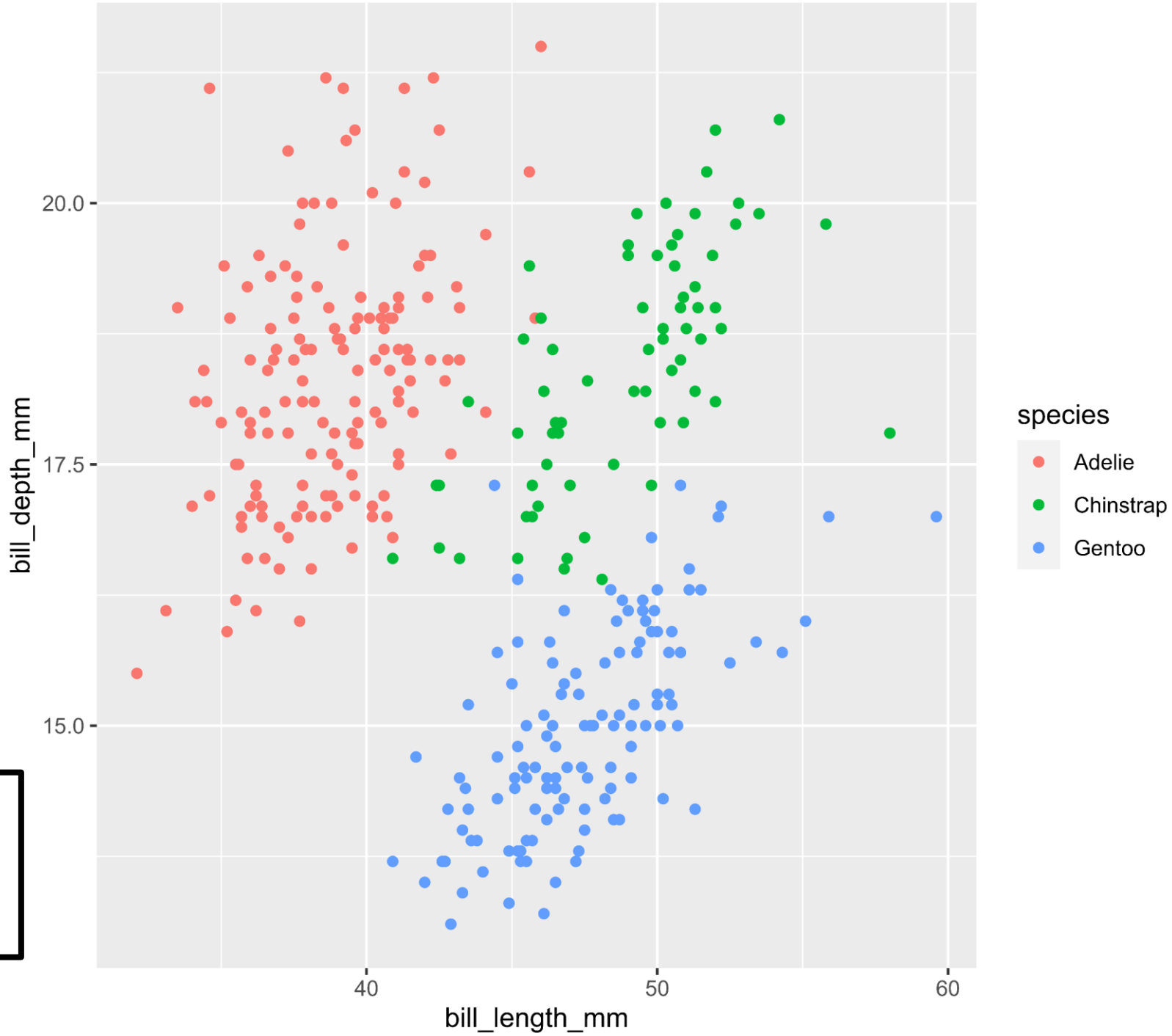
- morphological data for 3 penguin species
- example: bill depth vs bill length



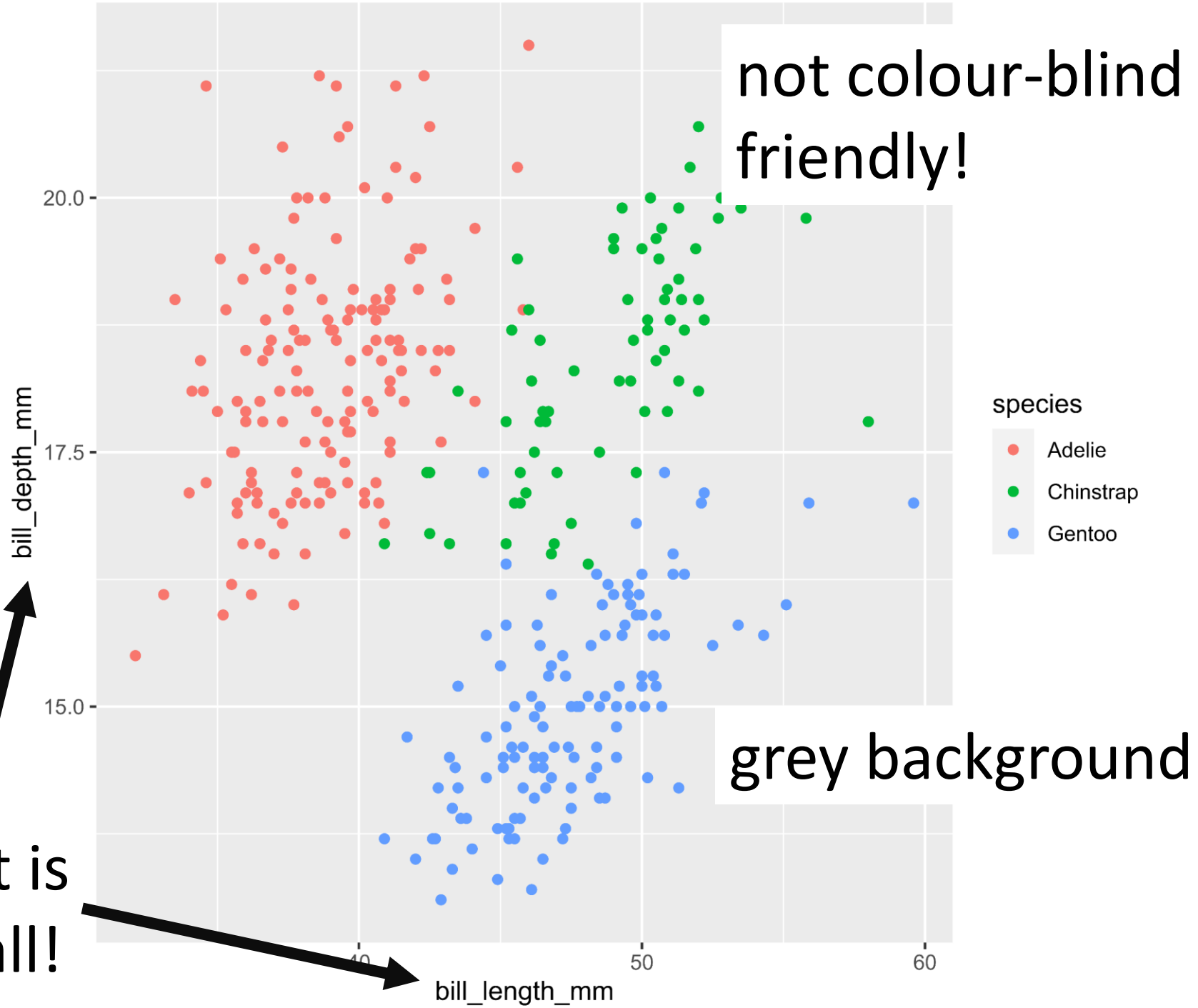

```
data(penguins)

ggplot() +
  geom_point(data = penguins,
             aes(x = bill_length_mm, y = bill_depth_mm, color = species)
```

DO NOT USE `ggplot()` defaults!

What do you think should be changed?



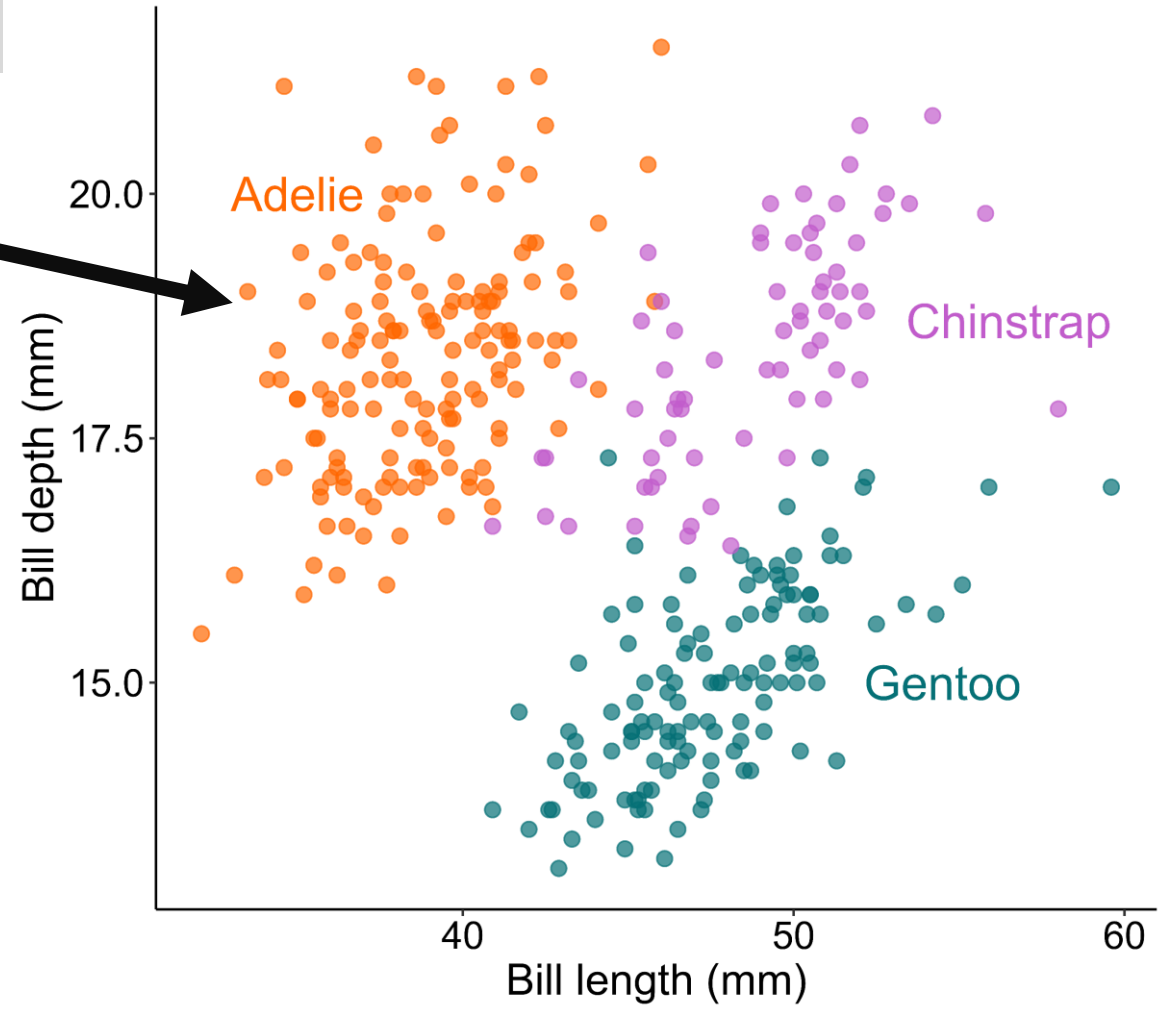
DO NOT USE
`ggplot()`
defaults!



```
ggplot() +
  geom_point(data = penguins,
             aes(x = bill_length_mm, y = bill_depth_mm, color = species)
```

```
geom_point(size = 2.7, alpha = 0.7)
```

increase point size

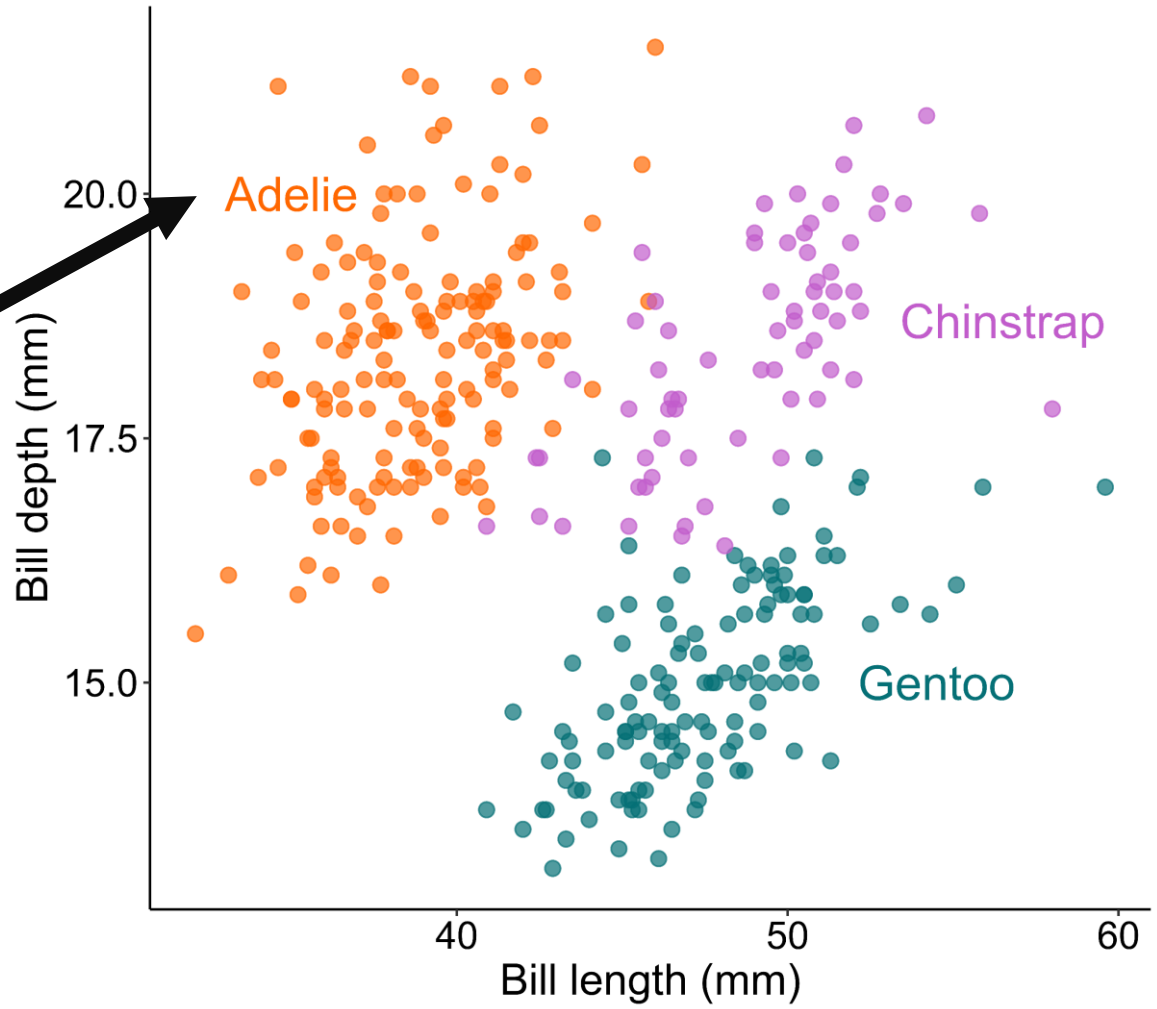


```
ggplot() +
  geom_point(data = penguins,
             aes(x = bill_length_mm, y = bill_depth_mm, color = species)
```

```
geom_point(size = 2.7, alpha = 0.7)
```

```
geom_text(aes(…, label = species,
                  color = species) +
theme(legend.position = “none)
```

species names on plot, instead of legend



```
ggplot() +
  geom_point(data = penguins,
             aes(x = bill_length_mm, y = bill_depth_mm, color = species)
```

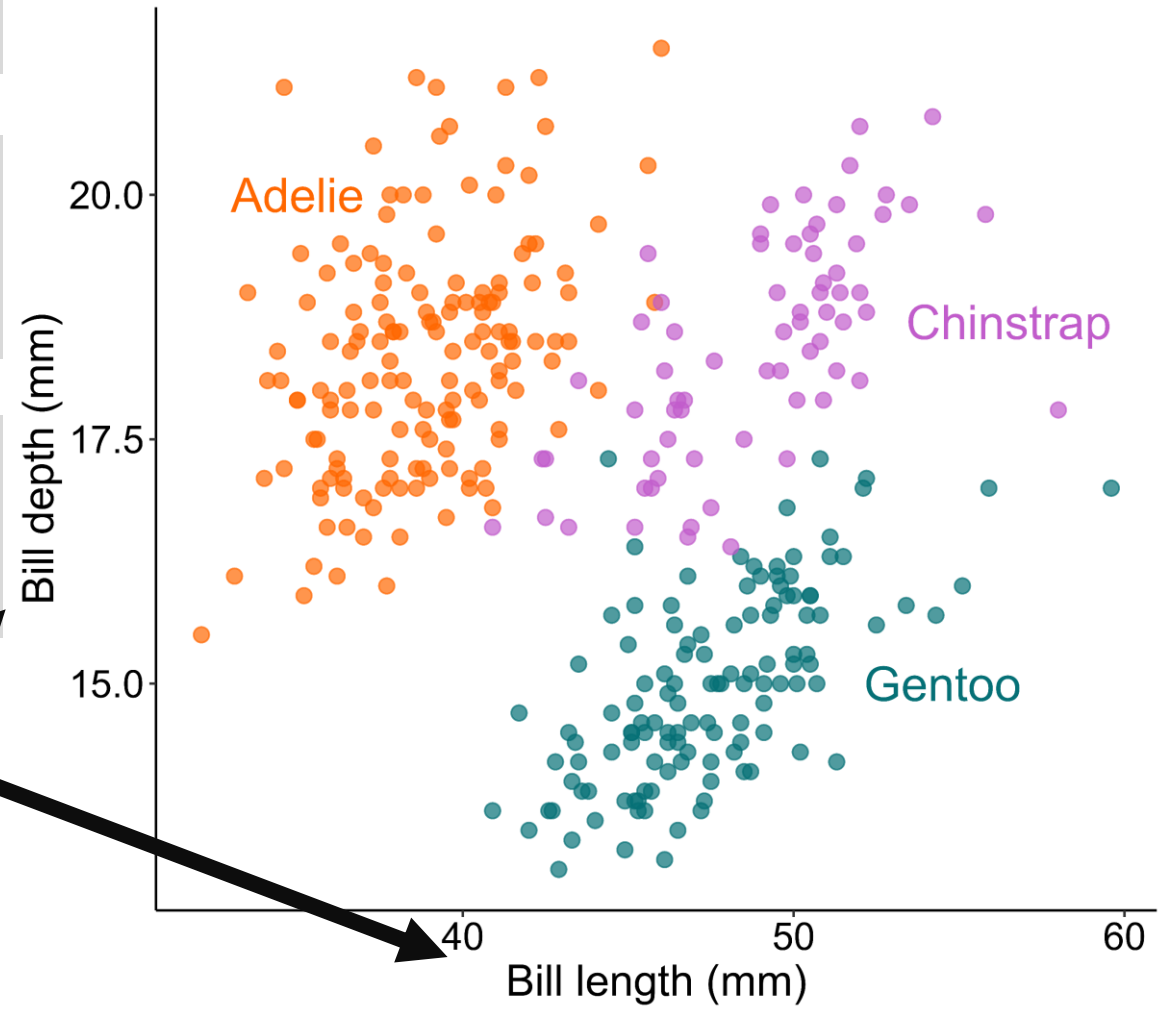
```
geom_point(size = 2.7, alpha = 0.7)
```

```
geom_text(aes(…, label = species,
                    color = species) +
theme(legend.position = “none)
```

```
theme_classic() +
theme(axis.title = element_text(size = 18,
      axis.text = element_text(size = 16))
```

remove grey background

increase axis text



```
ggplot() +
  geom_point(data = penguins,
             aes(x = bill_length_mm, y = bill_depth_mm, color = species)
```

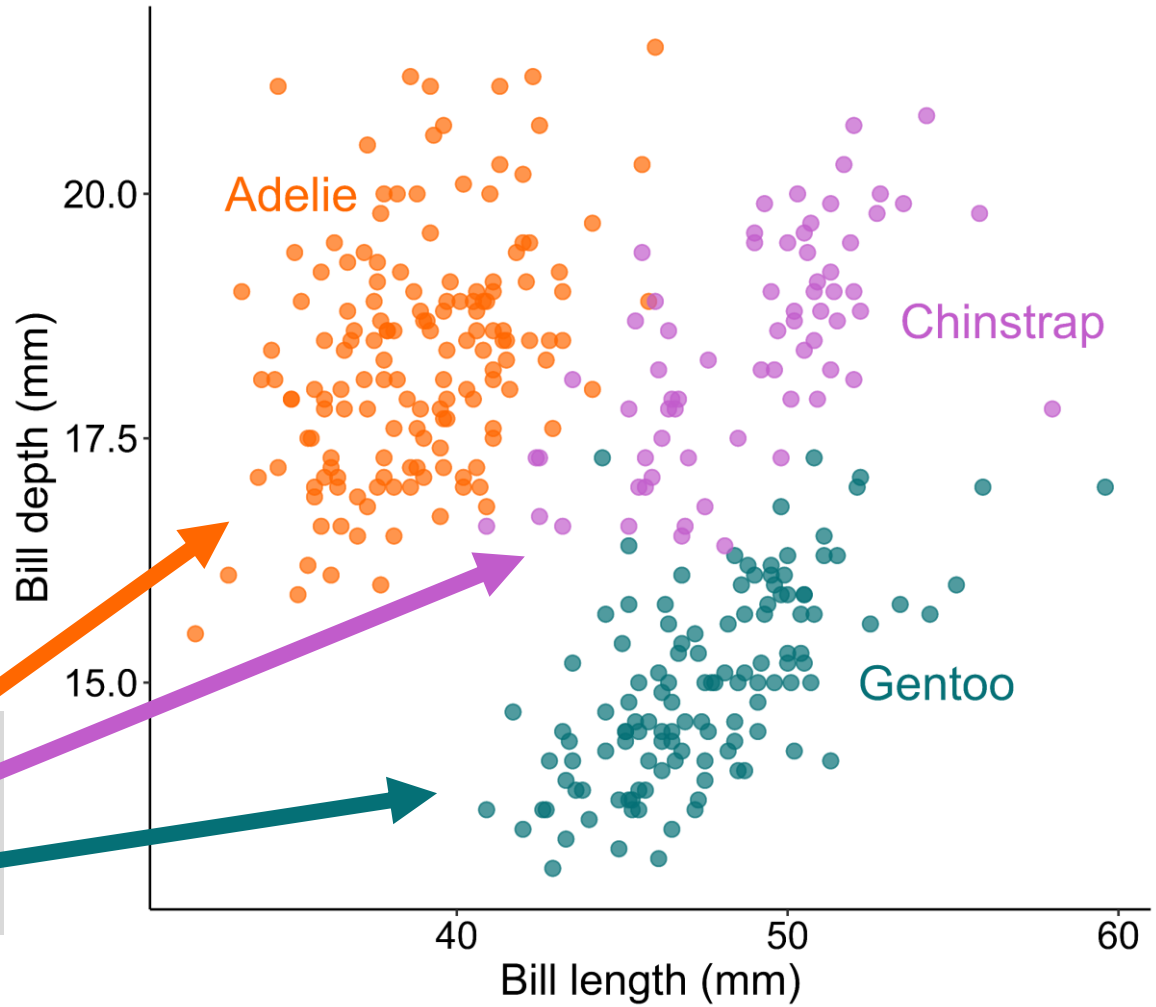
```
geom_point(size = 2.7, alpha = 0.7)
```

```
geom_text(aes(…, label = species,
                  color = species) +
theme(legend.position = “none)
```

```
theme_classic() +
theme(axis.title = element_text(size = 18,
      axis.text = element_text(size = 16))
```

```
scale_colour_manual(values = c(“#ff6700”,
                               “#c15ccb”,
                               “#057076”))
```

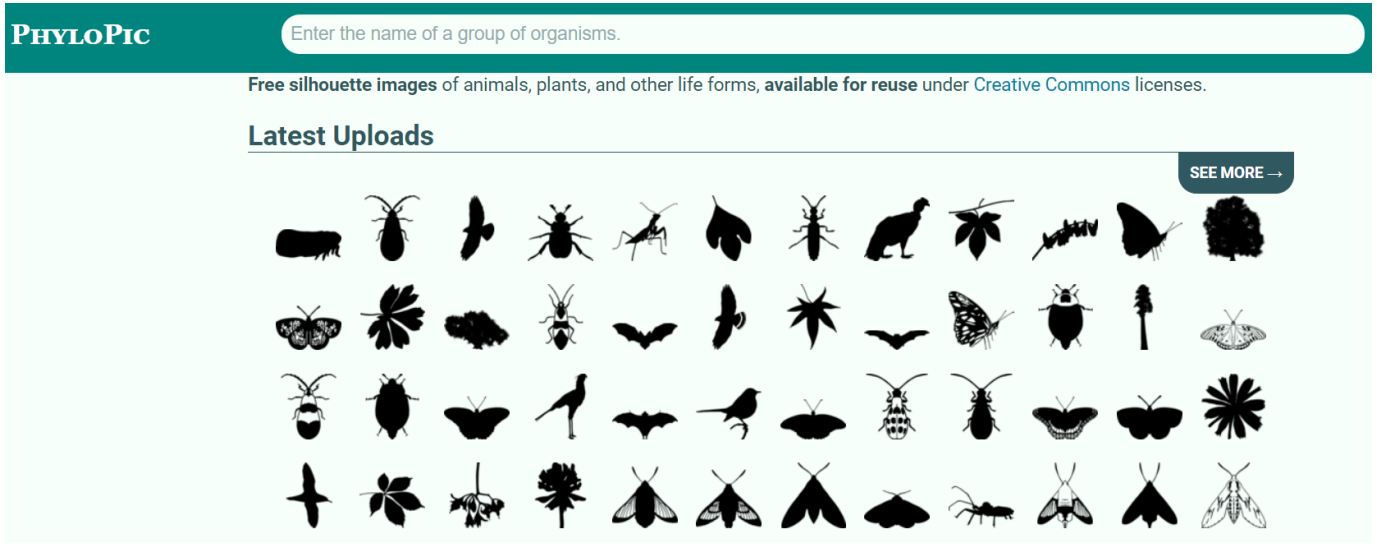
Custom colour palettes with hex codes

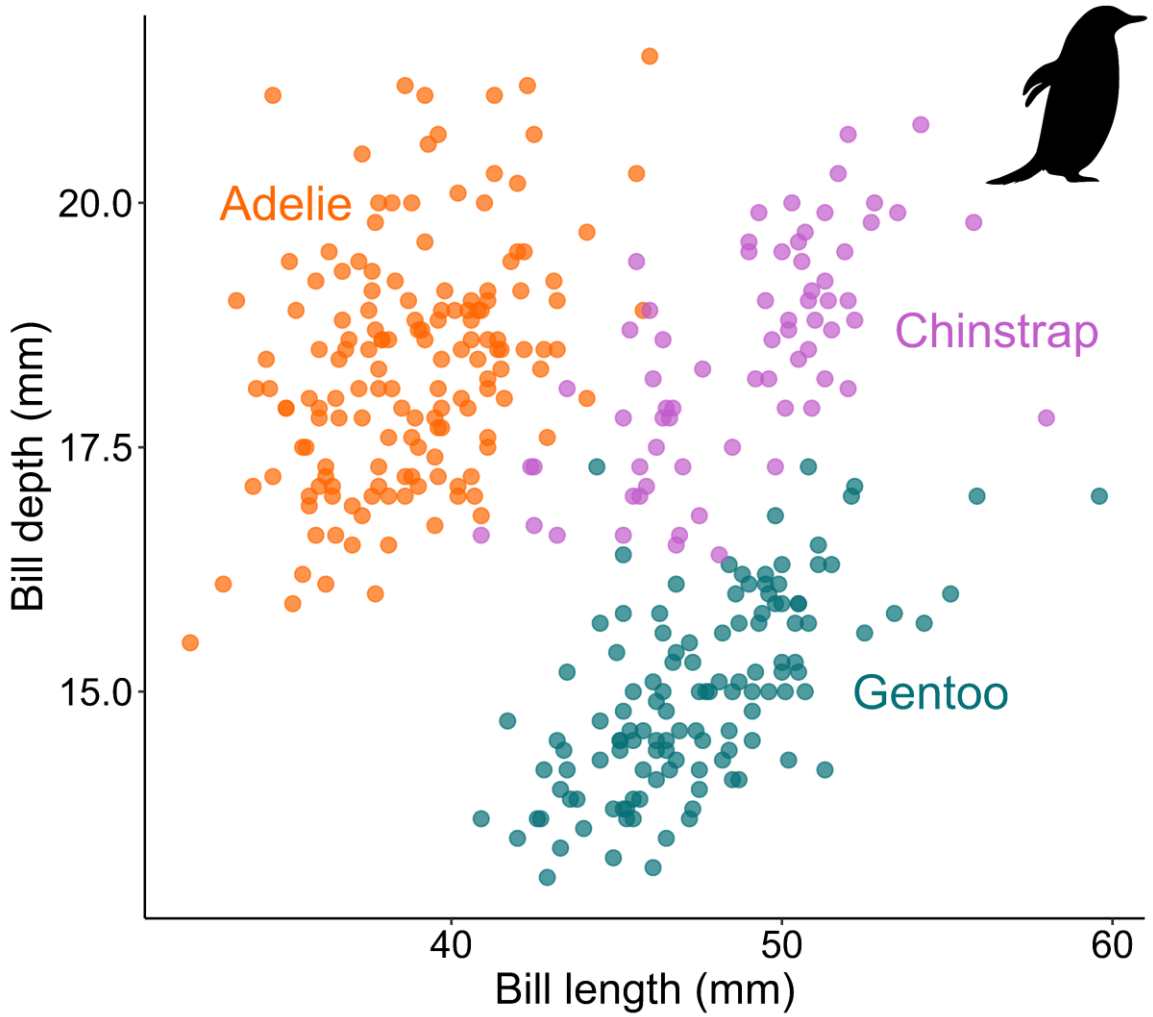


Adding organism silhouettes with `rphylopic`

Gentoo penguin scientific name

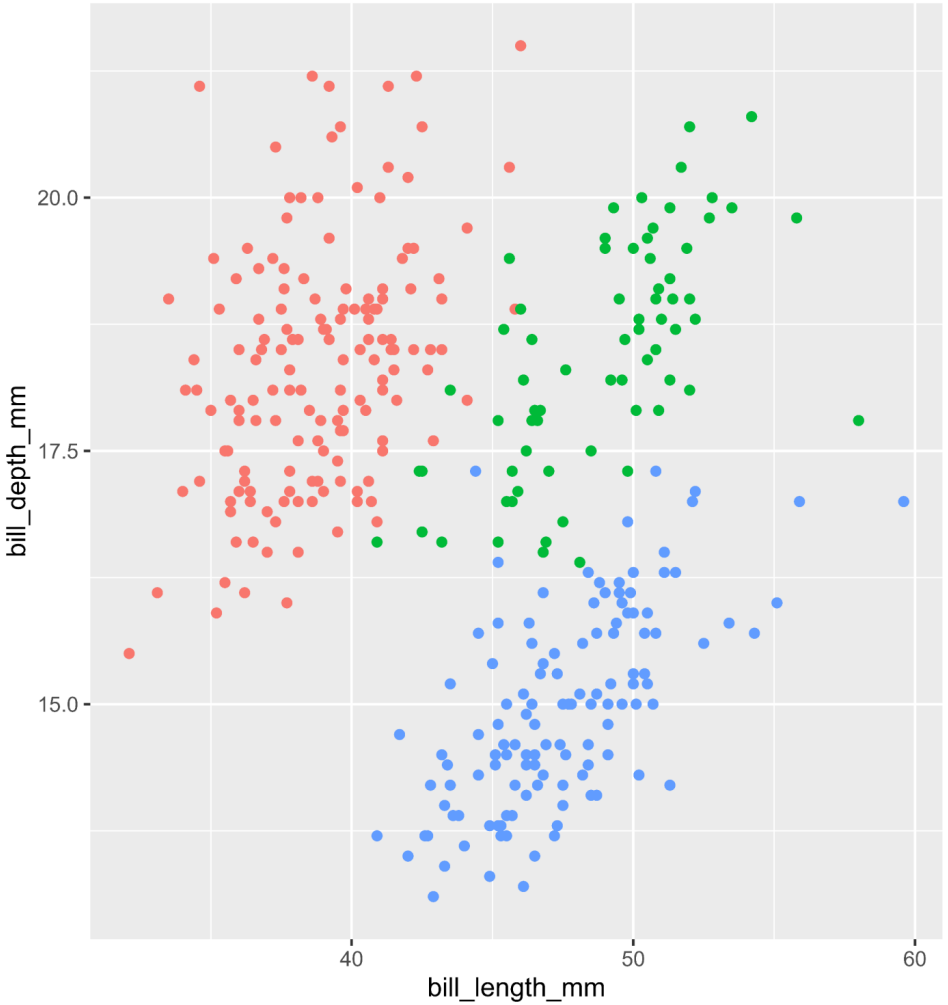
```
add_phylopic(name = "Pygoscelis papua,
             x = 58.6, y = 21.1, ysize = 6)
```



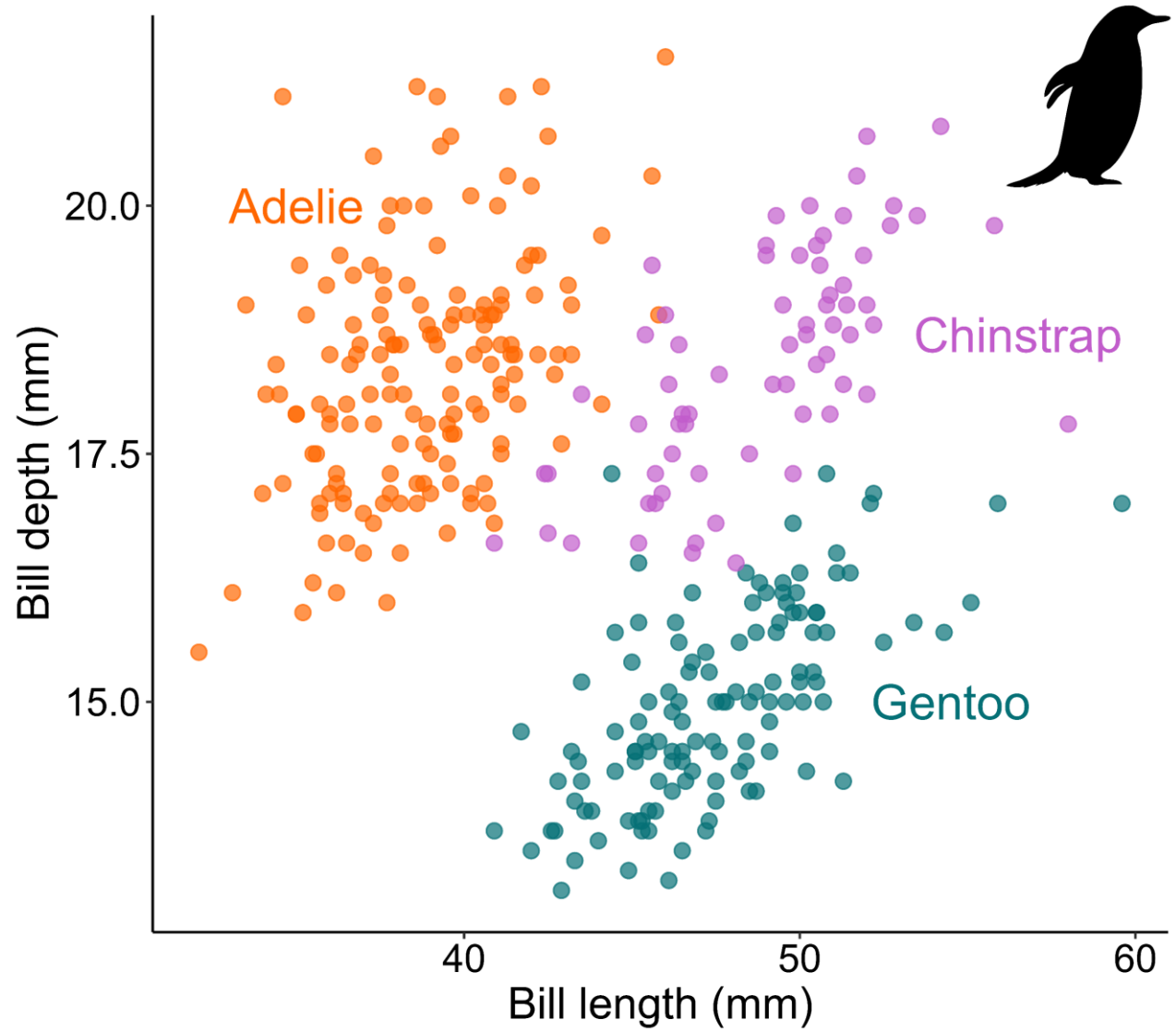


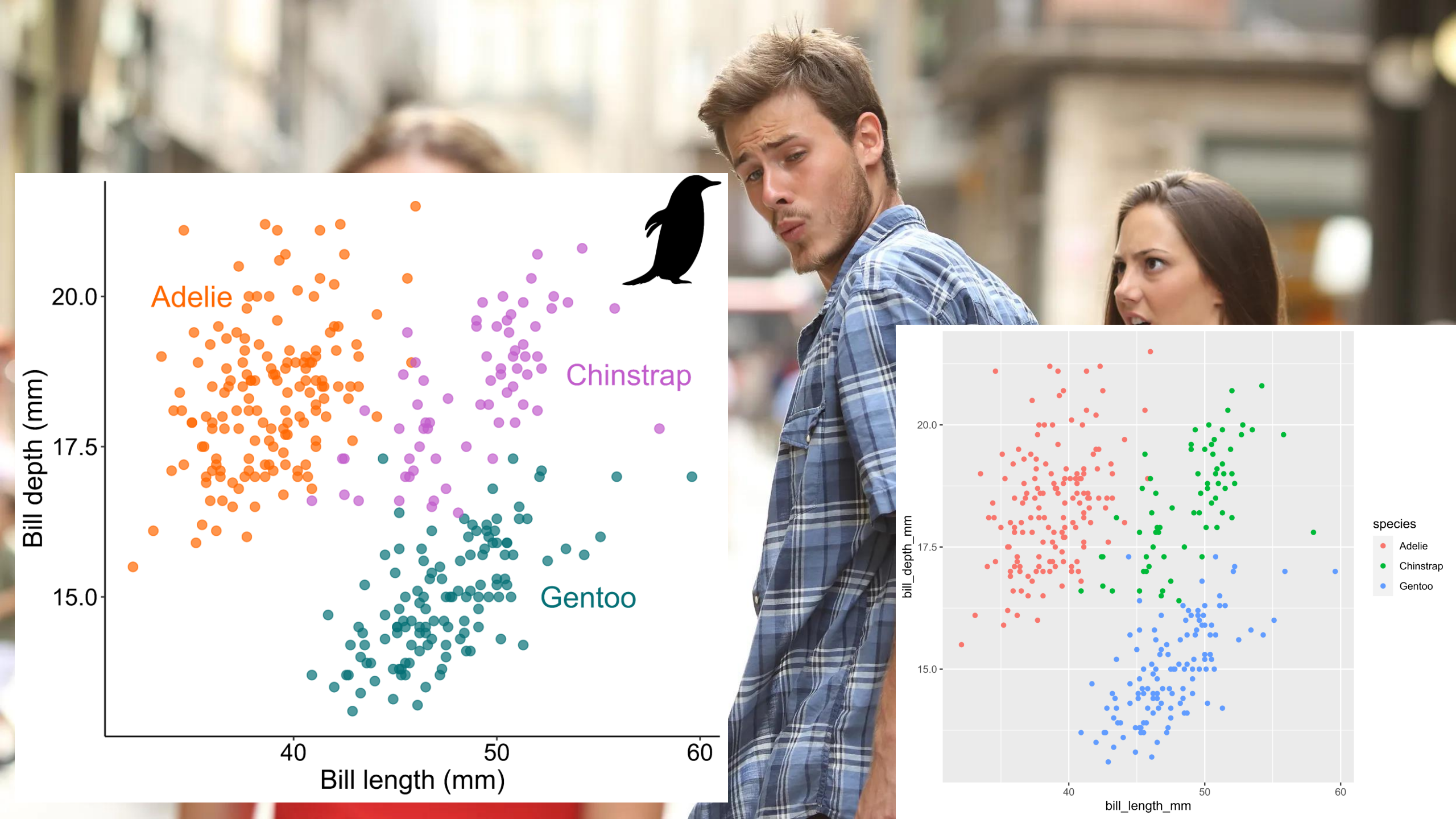
https://www.phylopic.org/

Going beyond `ggplot()` defaults is easy and makes a big difference!



vs



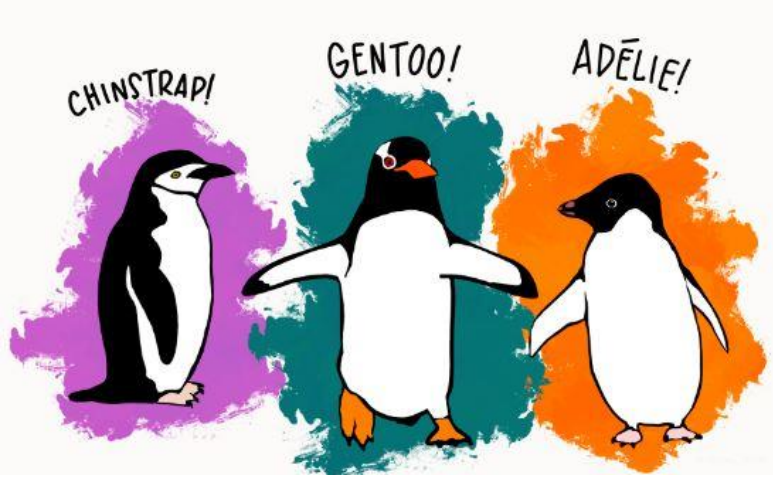

Adelie
Chinstrap
Gentoo
Bill depth (mm)
20.0
17.5
15.0
40
50
60
Bill length (mm)
bill_depth_mm
bill_length_mm
species
Adelie
Chinstrap
Gentoo

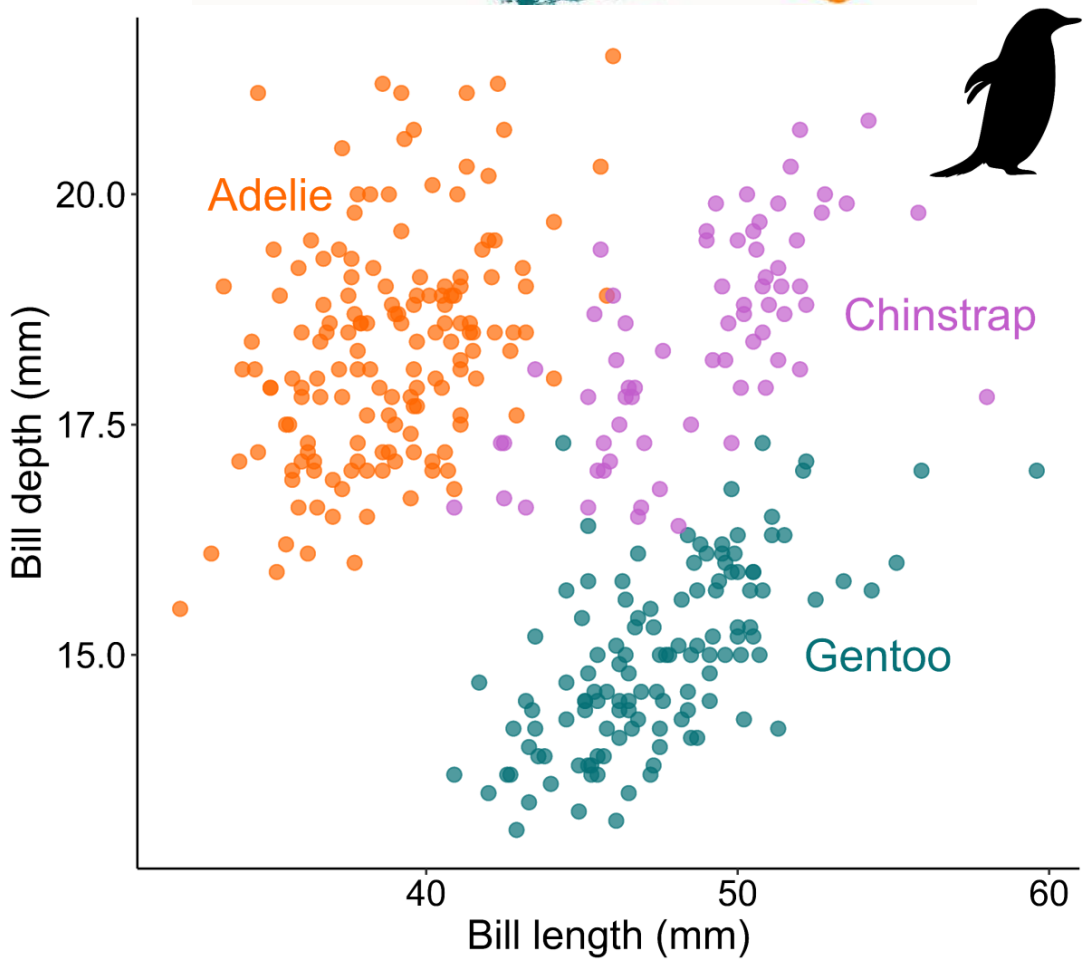
`ggplot()` summary

Different geoms: points, lines, polygons, text, etc

Do not use `ggplot()` defaults!

Have fun/be creative with `ggplot()`!



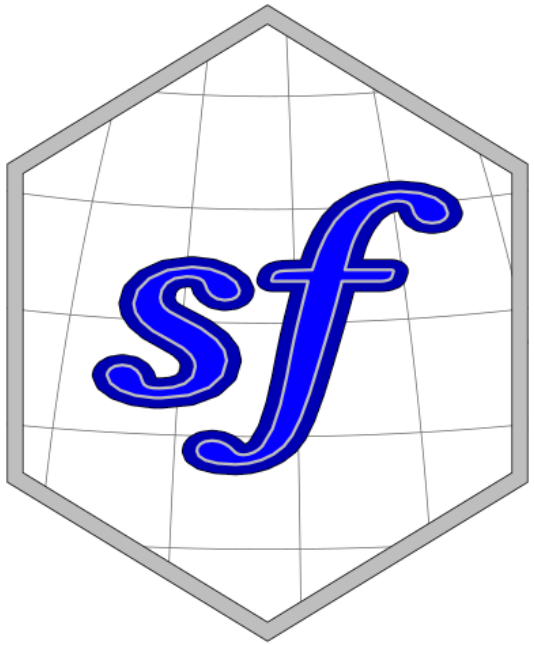


Additional resources

Hadley Wickham. R for data science
https://r4ds.had.co.nz/data-visualisation.html
Hadley Wickham. ggplot2: Elegant graphics for data analysis
https://ggplot2-book.org/

Back to maps!

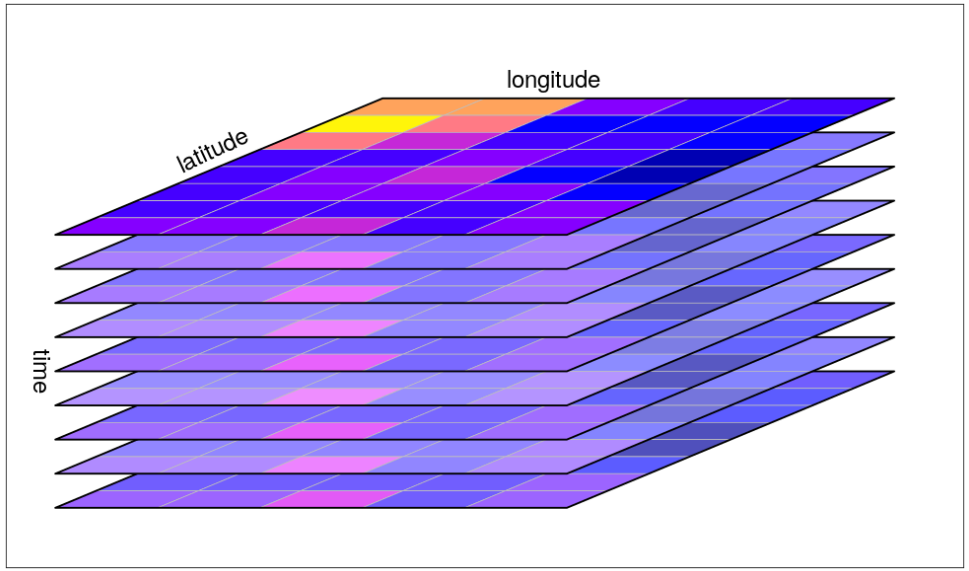
Which R packages to use?



vectors

rasters

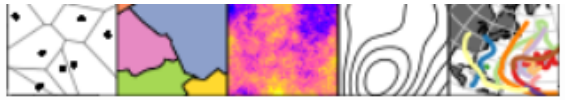
stars



“raster cubes” and other vector/raster spatiotemporal arrays

e.g., satellite data (composed of multiple bands at different wavelengths)

r-spatial



Events Projects Book About SDSL

R-spatial evolution: retirement of rgdal, rgeos and maptools

Apr 12, 2022 • Edzer Pebesma, Roger Bivand

[view raw Rmd]

Summary: Packages `rgdal`, `rgeos` and `maptools` will retire by the end of 2023 . We describe where their functionality will go, what package maintainers can or should do, and which steps we will take to minimize the impact on dependent packages and on reproducibility in general.

https://r-spatial.org/r/2022/04/12/evolution.html

Built in basemaps in R

`RNaturalEarth` package

coarse resolution, great for global maps, bad for regional/local maps

`OpenStreetMap` package

load in satellite and other basemaps

`basemaps` package

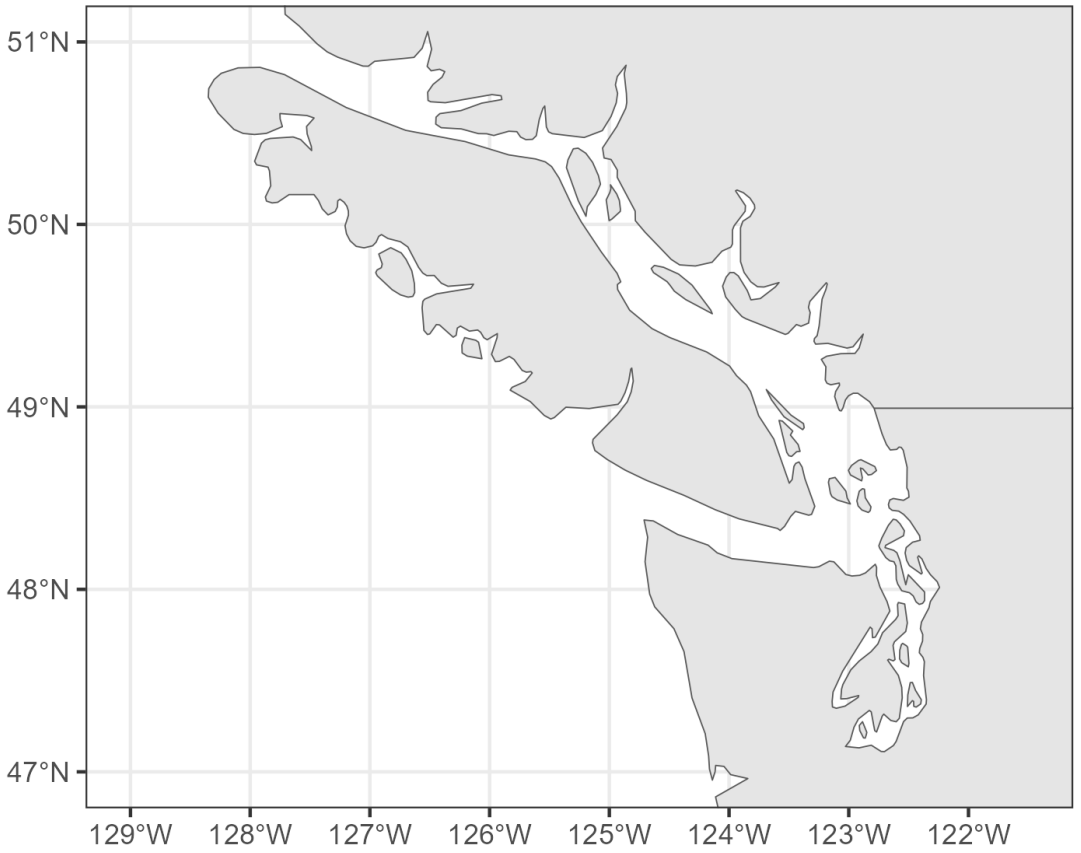
satellite and other basemaps

Can also load in your own basemaps with `geom_sf()`

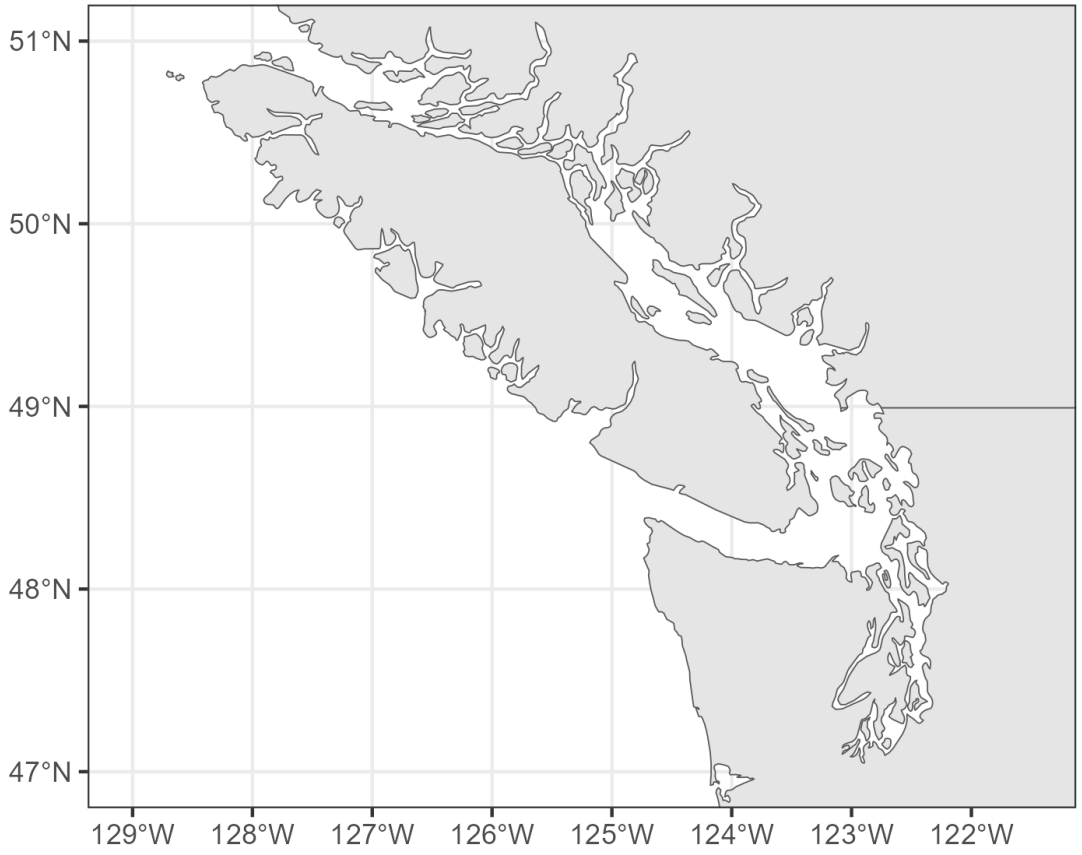
RNaturalEarth

Includes different resolutions: **1:10M is best for regional-scale maps**

RNaturalEarth 1:50M



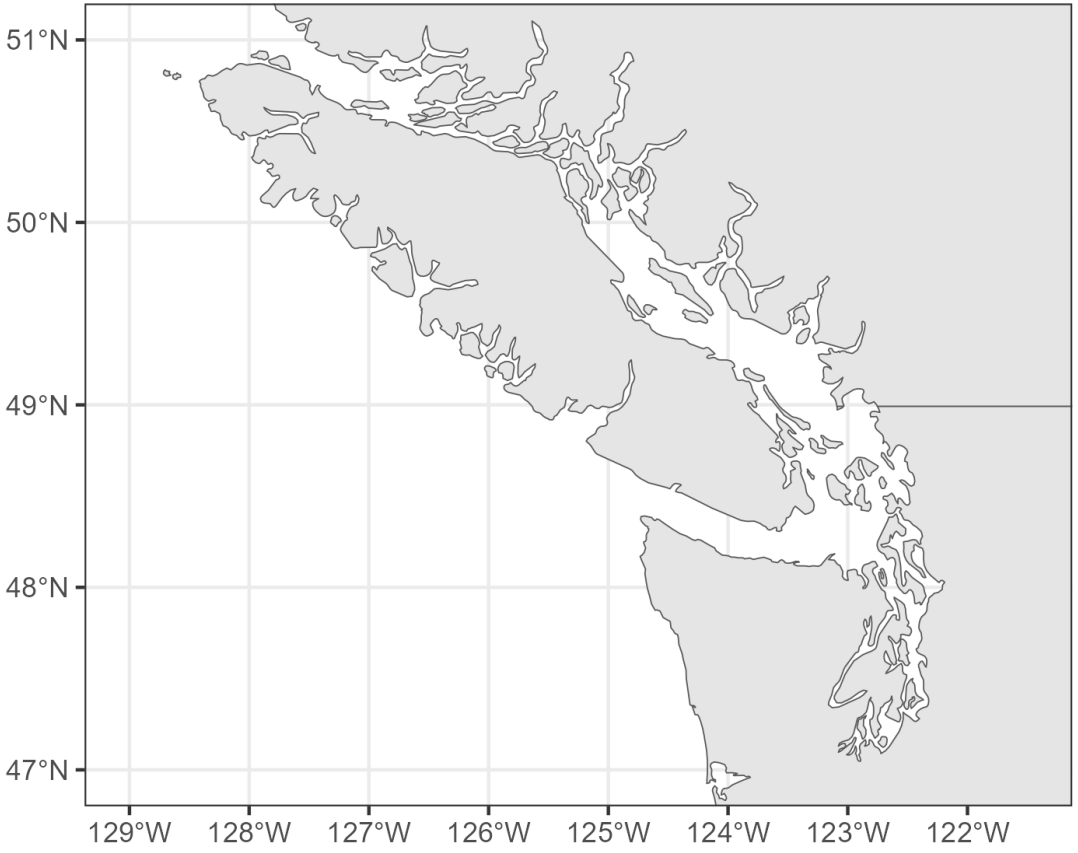
RNaturalEarth 1:10M



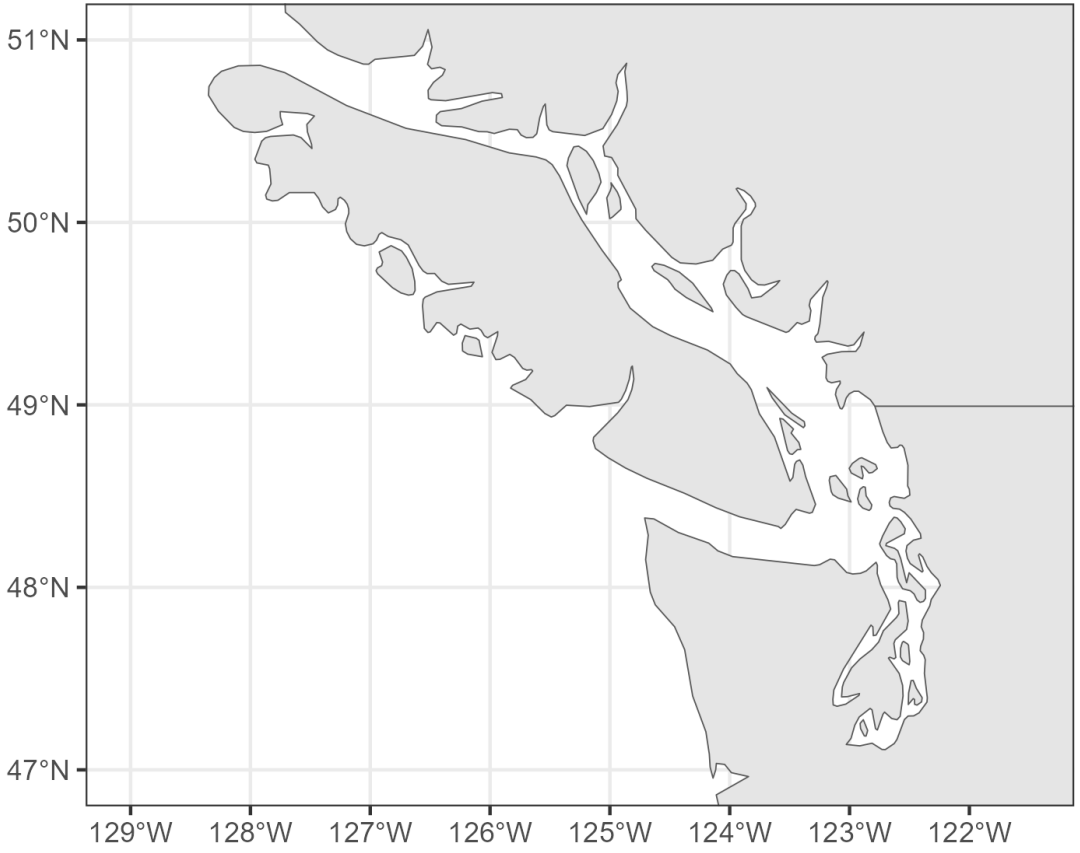
RNaturalEarth

Includes different resolutions: **1:10M is best for regional-scale maps**

RNaturalEarth 1:10M



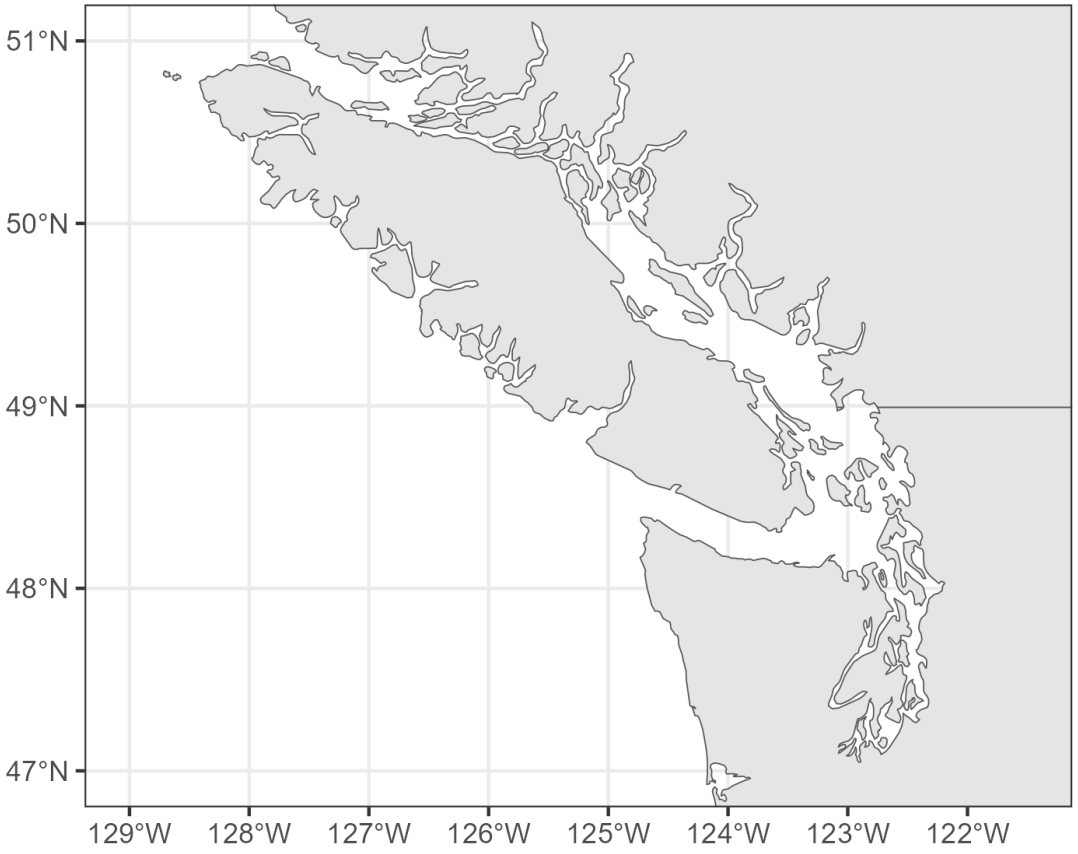
RNaturalEarth 1:50M



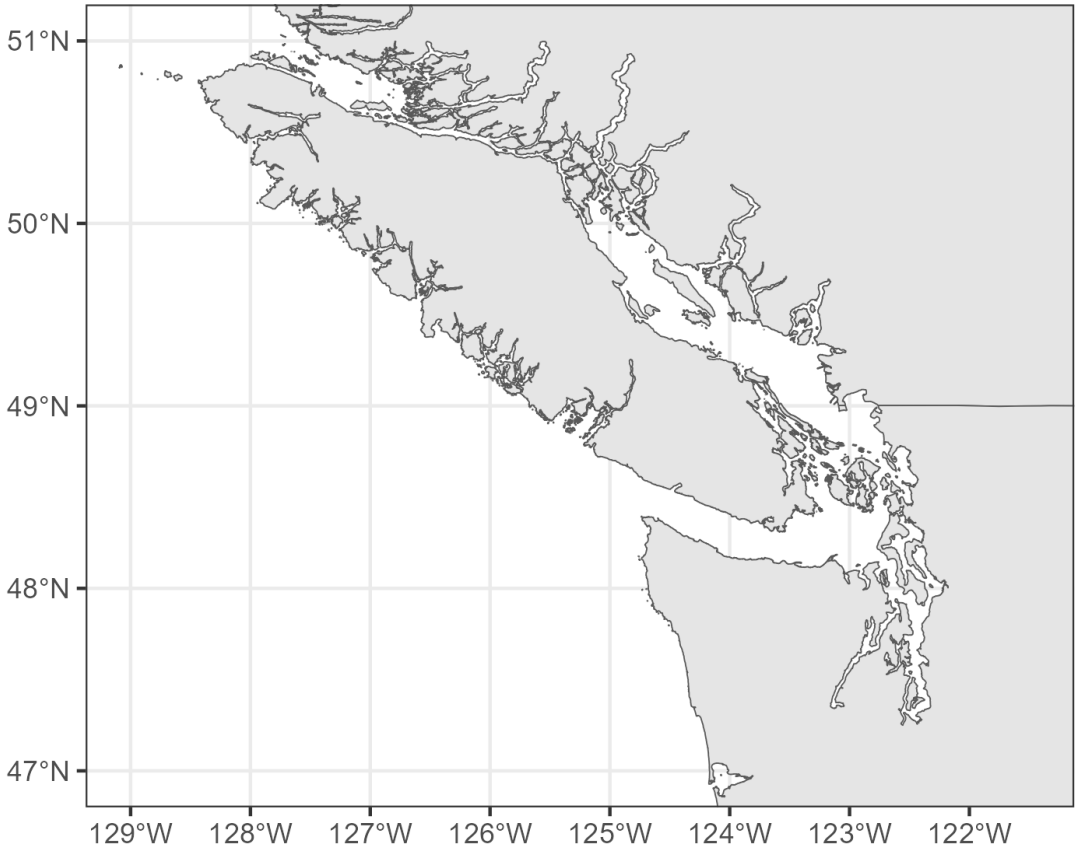
Compare RNaturalEarth to bc-coast.shp

bc-coast.shp: shapefile provided in data folder of Github repository

RNaturalEarth 1:10M

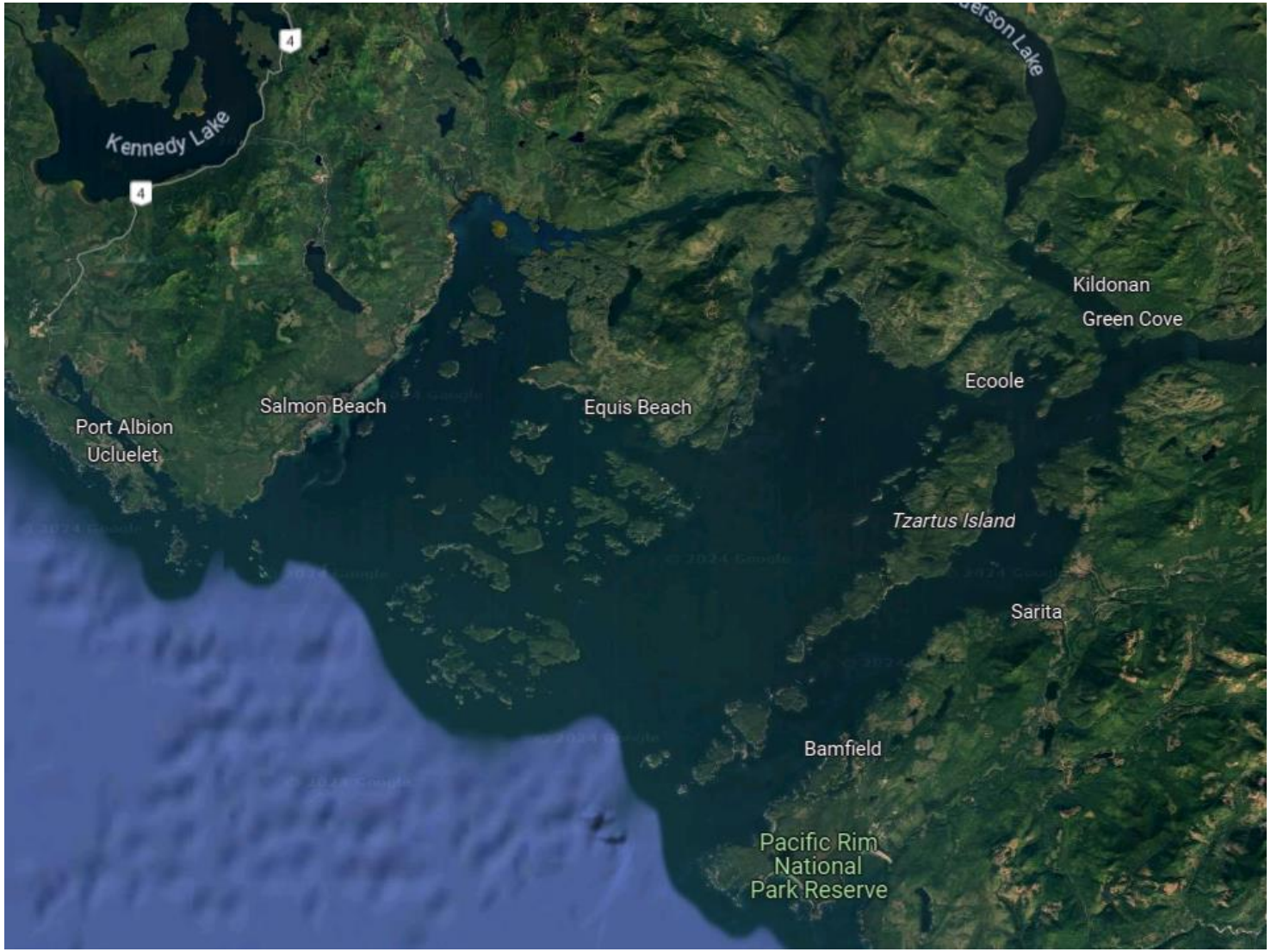


BC coast shapefile



Compare RNaturalEarth to bc-coast.shp

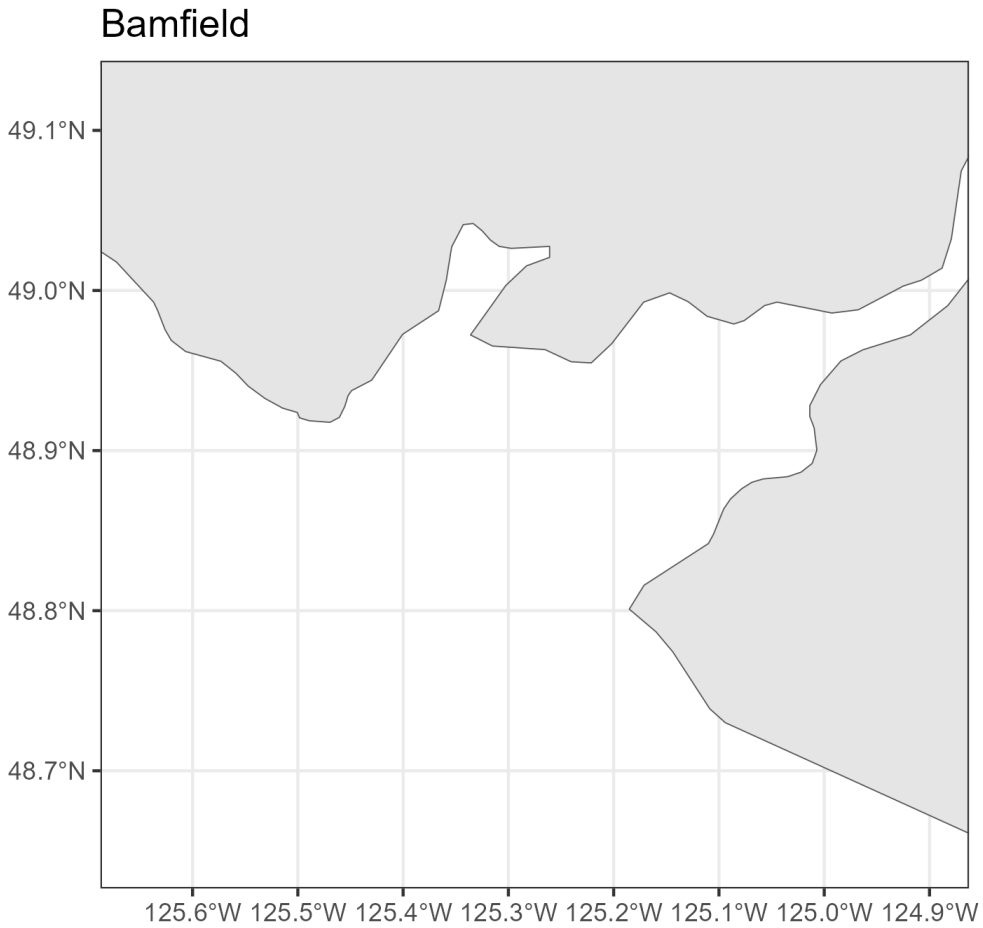
Bamfield, west coast of Vancouver Island



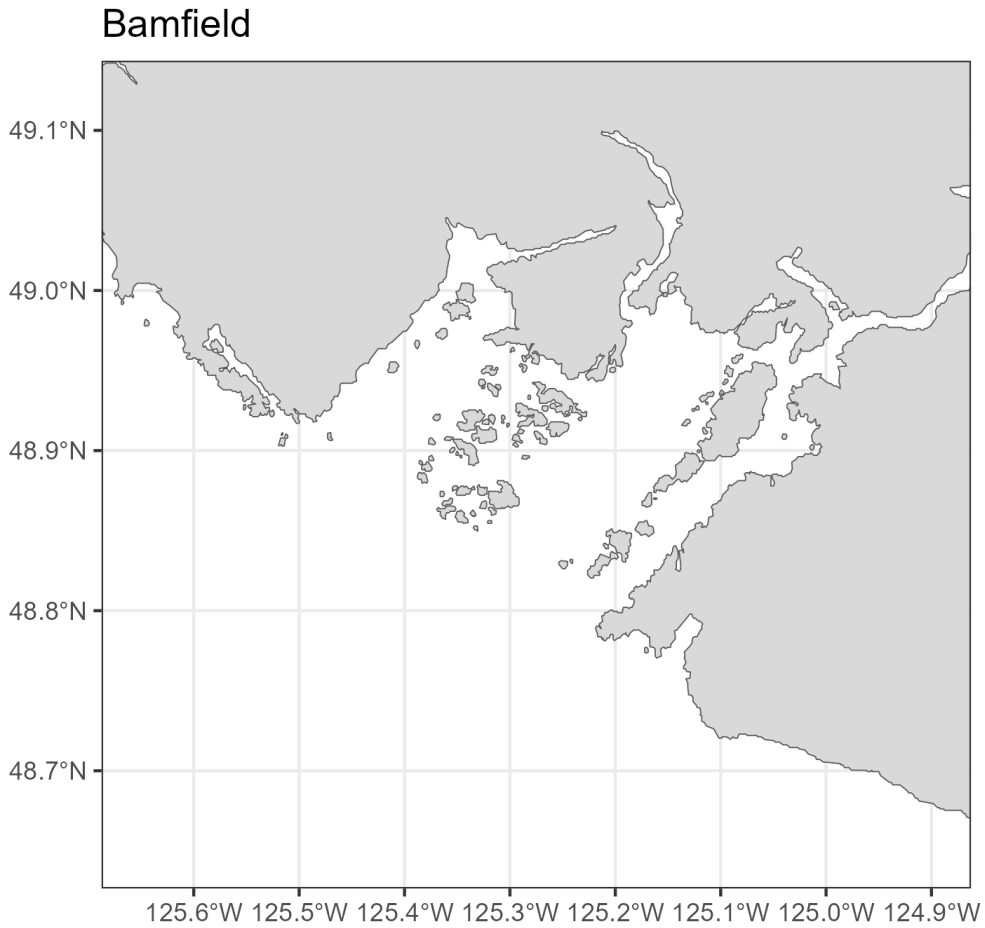
Compare RNaturalEarth to bc-coast.shp

Bamfield, west coast of Vancouver Island

RNaturalEarth 1:10M

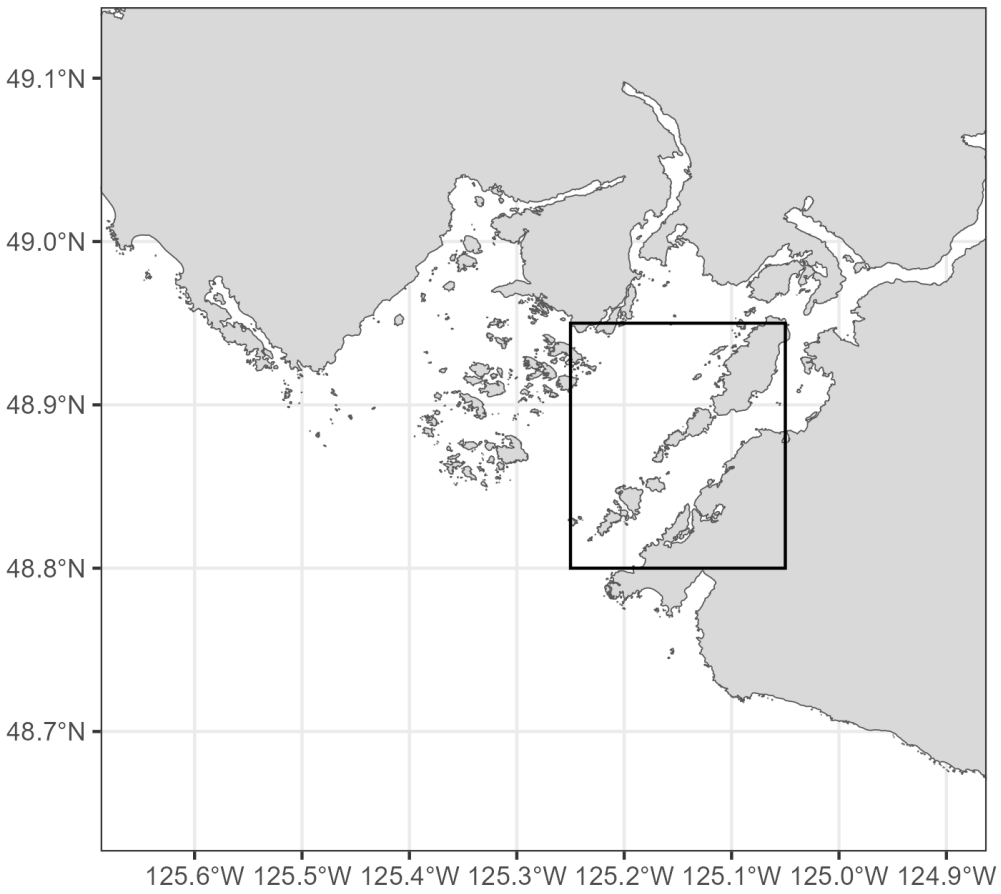


BC coast shapefile

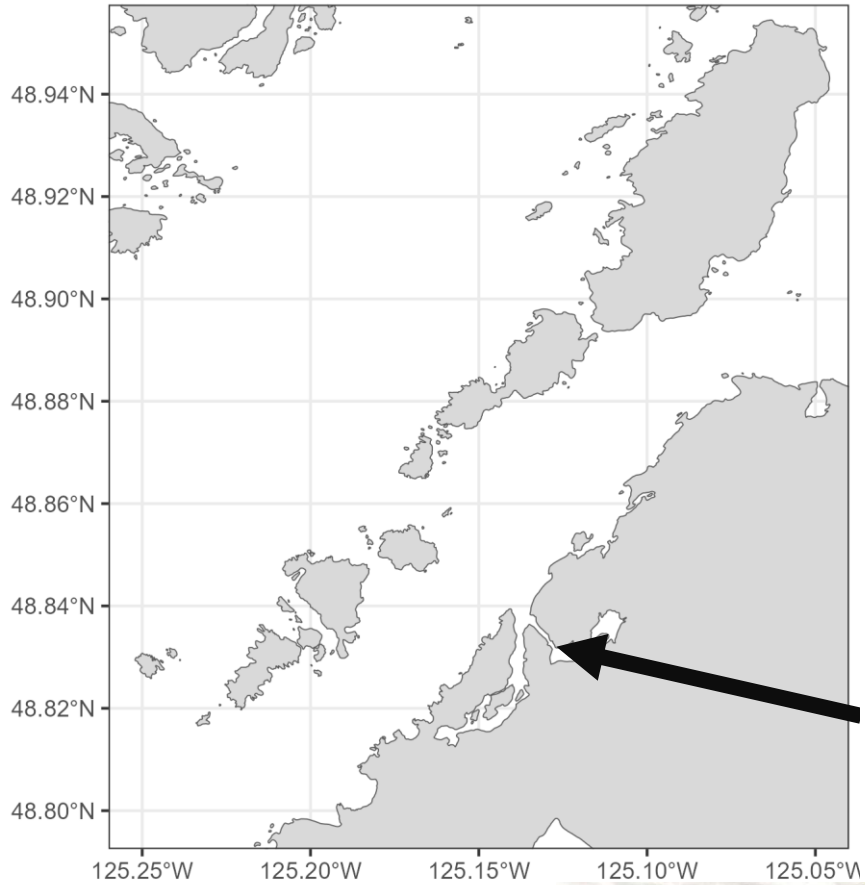


Higher-resolution for local maps

Even higher resolution is available with shapefiles from Hakai



Zoom in



bmsc
BAMFIELD MARINE SCIENCES CENTRE

Hakai
Science on the Coastal Margin

https://hecate.hakai.org/rguide/mapping-in-r.html

Sneak peak: satellite maps in R

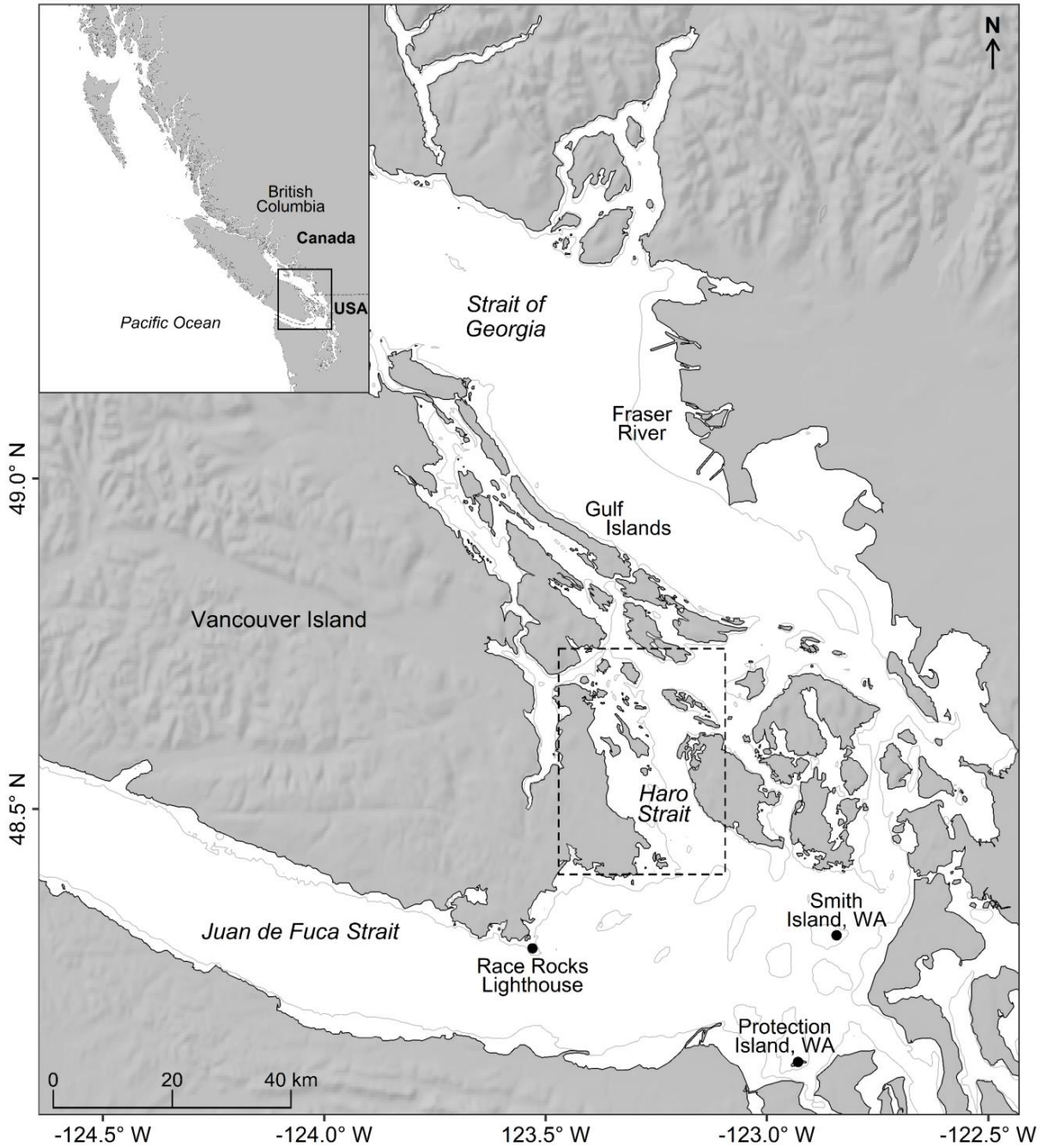
Even higher resolution with satellite maps

Zoom in

bmsc
BAMFIELD MARINE SCIENCES CENTRE

Study area maps

Goal: show sampling locations and important geographic/biological features



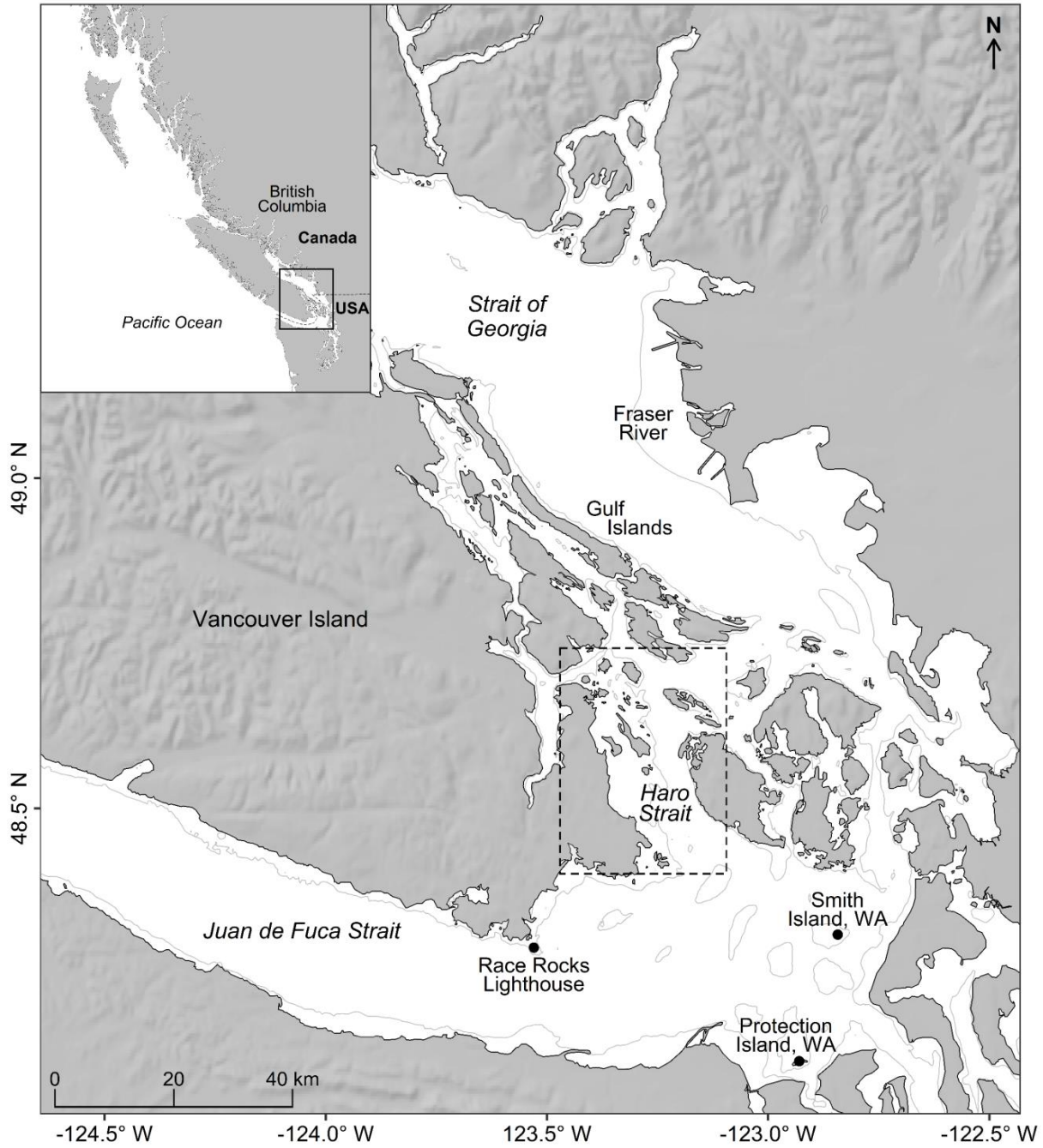
Robinson et al. 2023

Study area maps

Goal: show sampling locations and important geographic/biological features

Often greyscale for simplicity

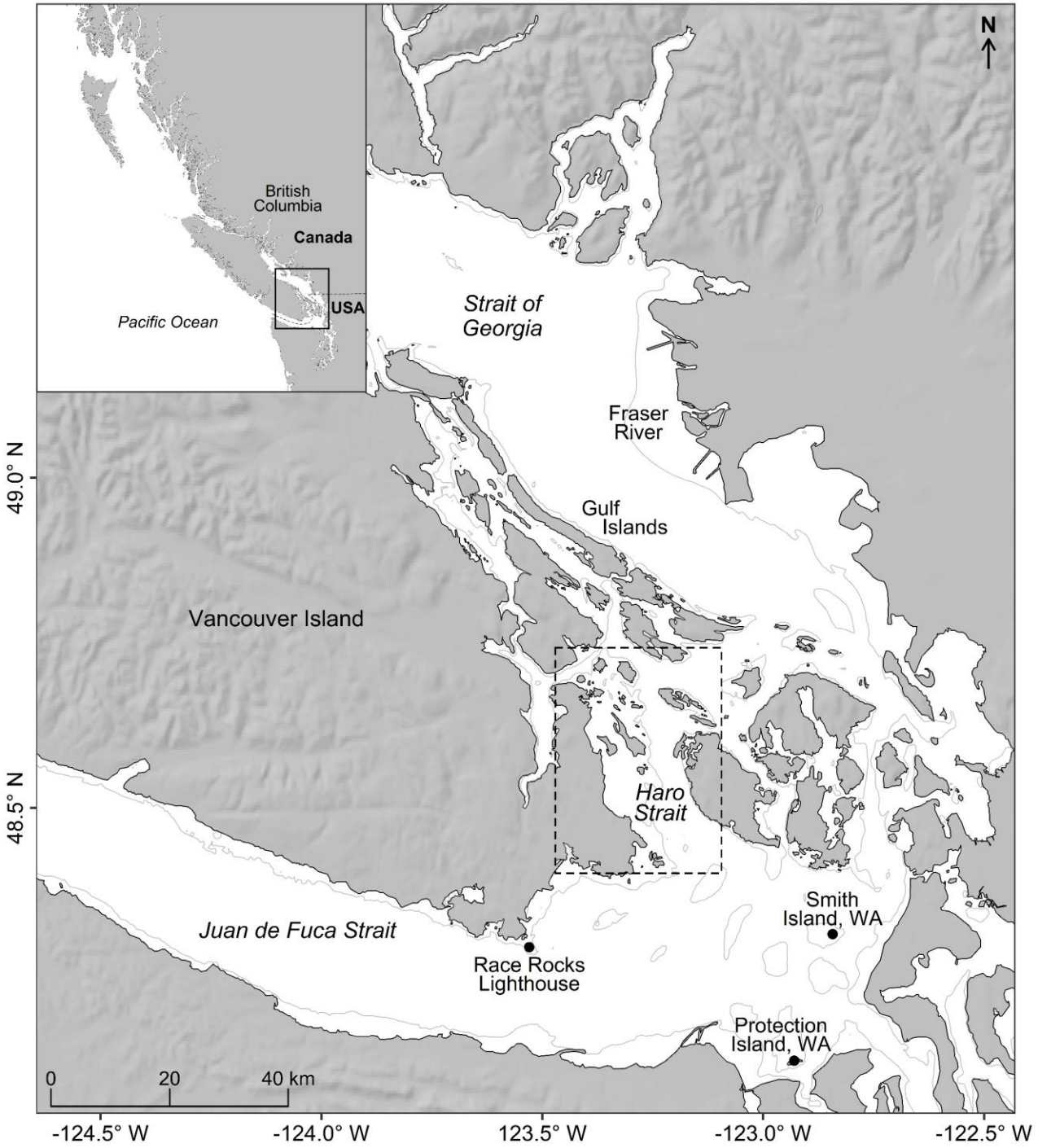
Include inset map showing larger geographic region



Robinson et al. 2023

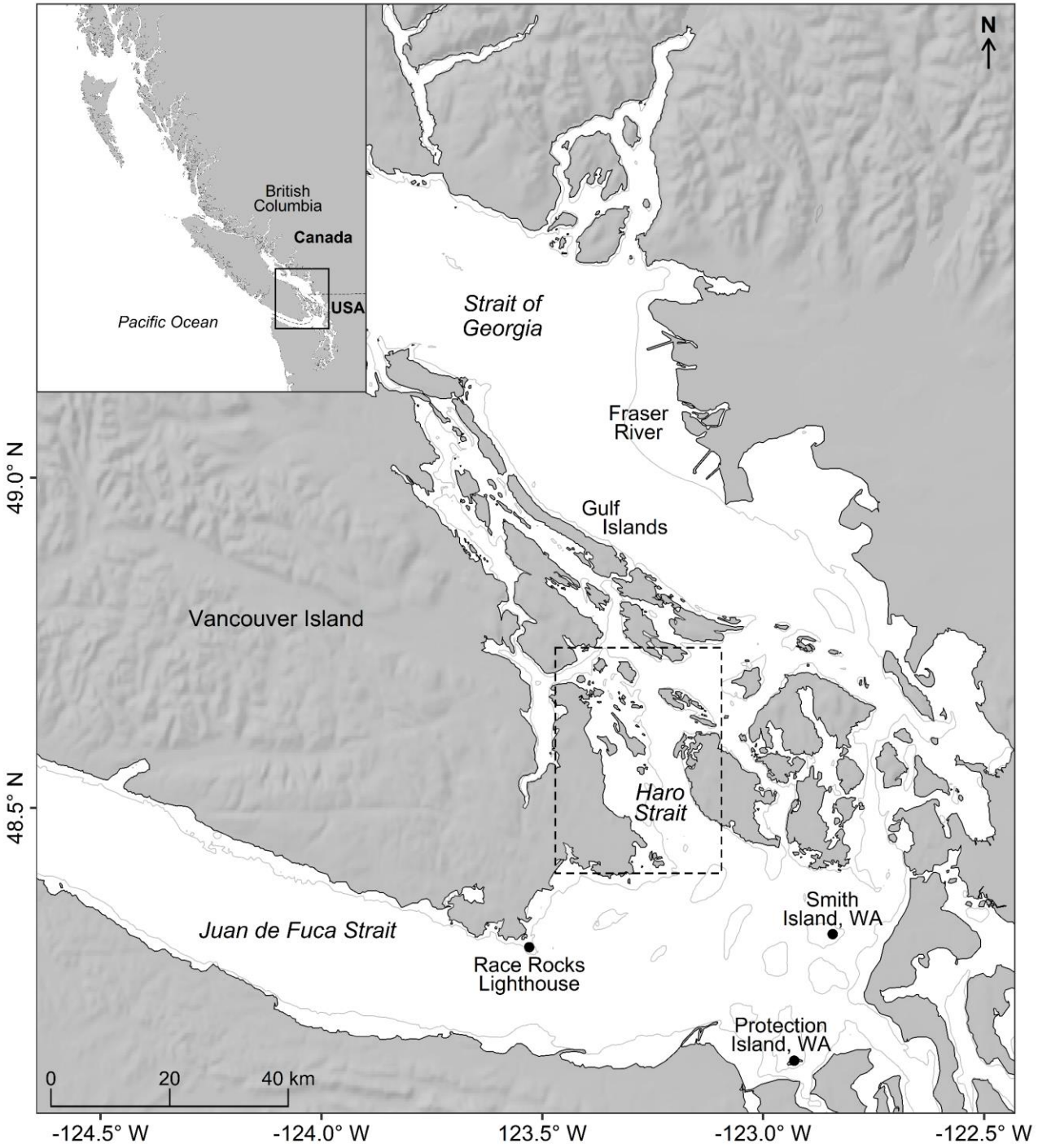
Important parts

Good spatial resolution is key



Important parts

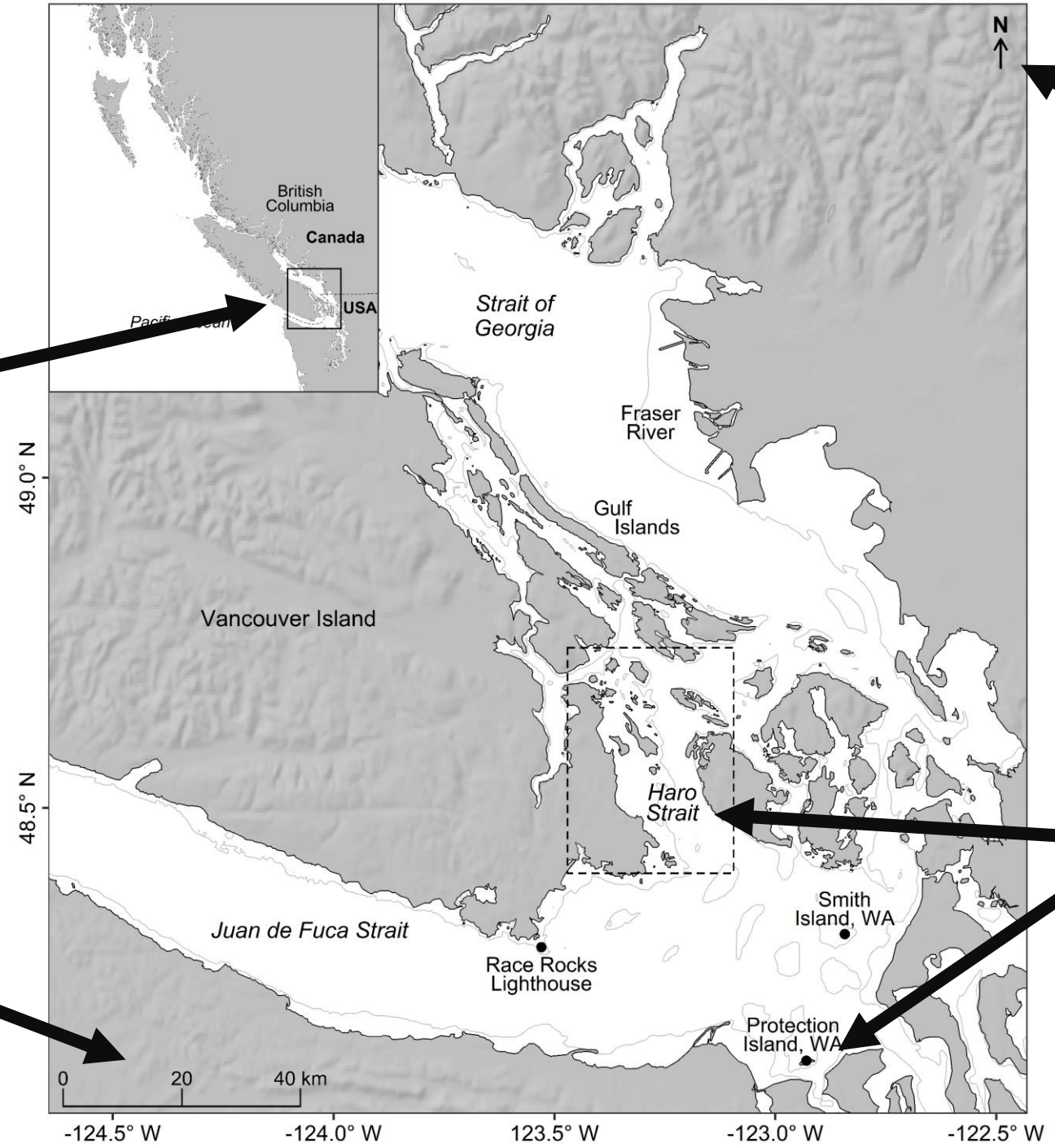
What do you think is important?



Important parts

Inset map showing where the study area is

Study area denoted by rectangle

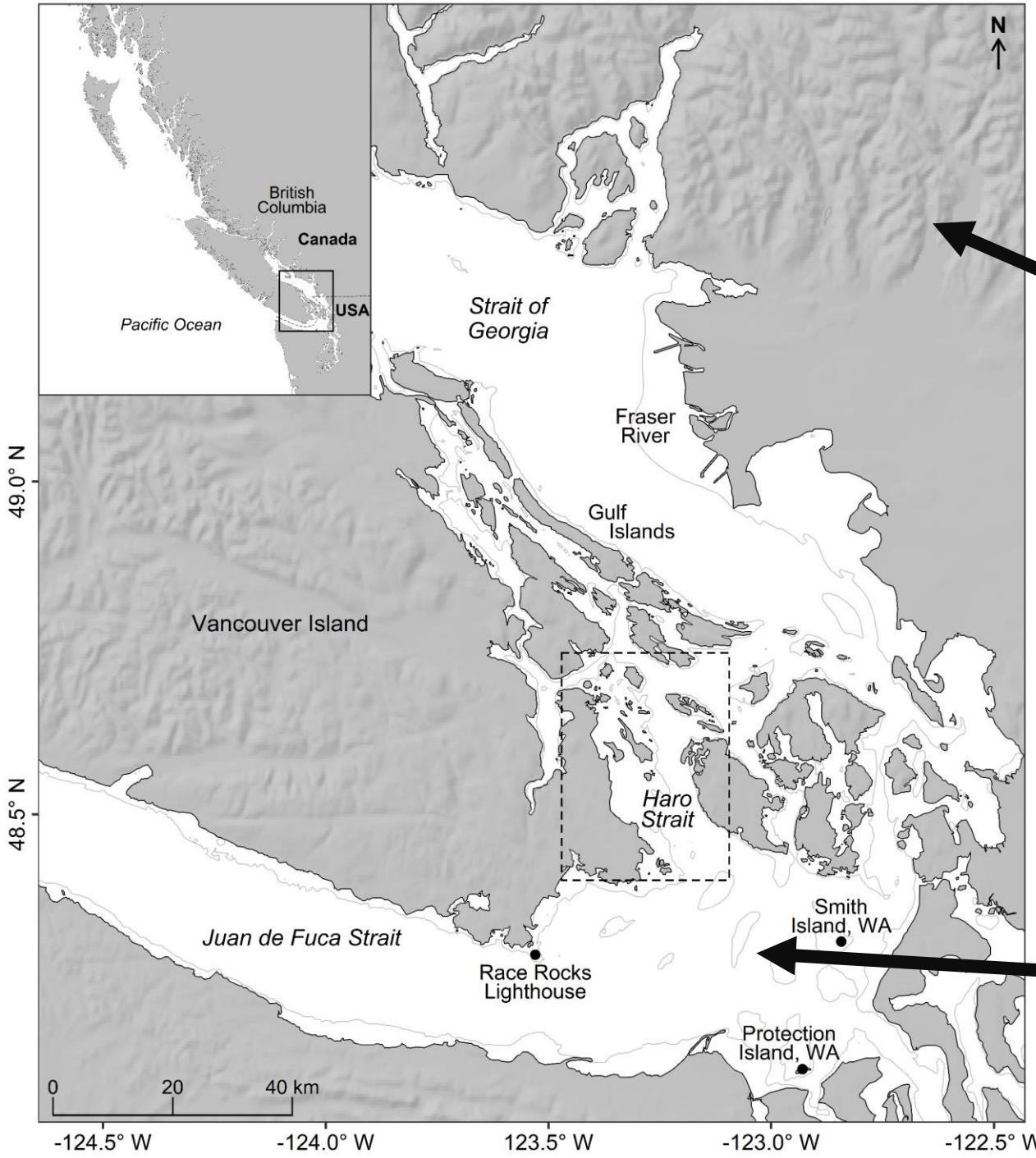


North arrow

Important sites labelled

Scale bar

Extra pieces



Hillshading to illustrate topography

50 metre depth contours

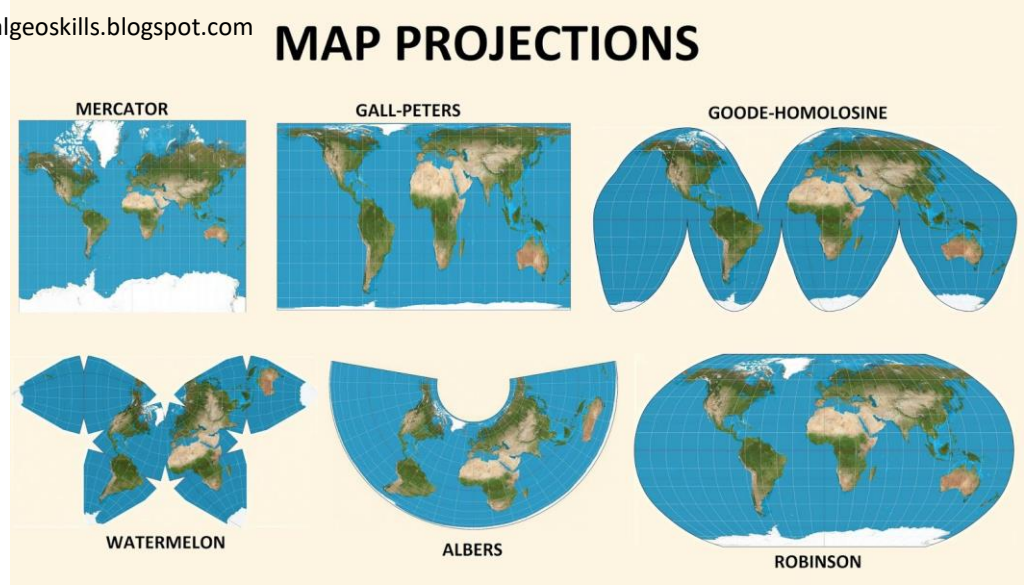
Projections

What projection to use?

What is a projection?

Projections are used to map the Earth's 3D surface on a 2D plane (the map)

Either in degrees (e.g., WGS 84) or metres/kilometres (e.g., BC Albers, UTM Zone 10)



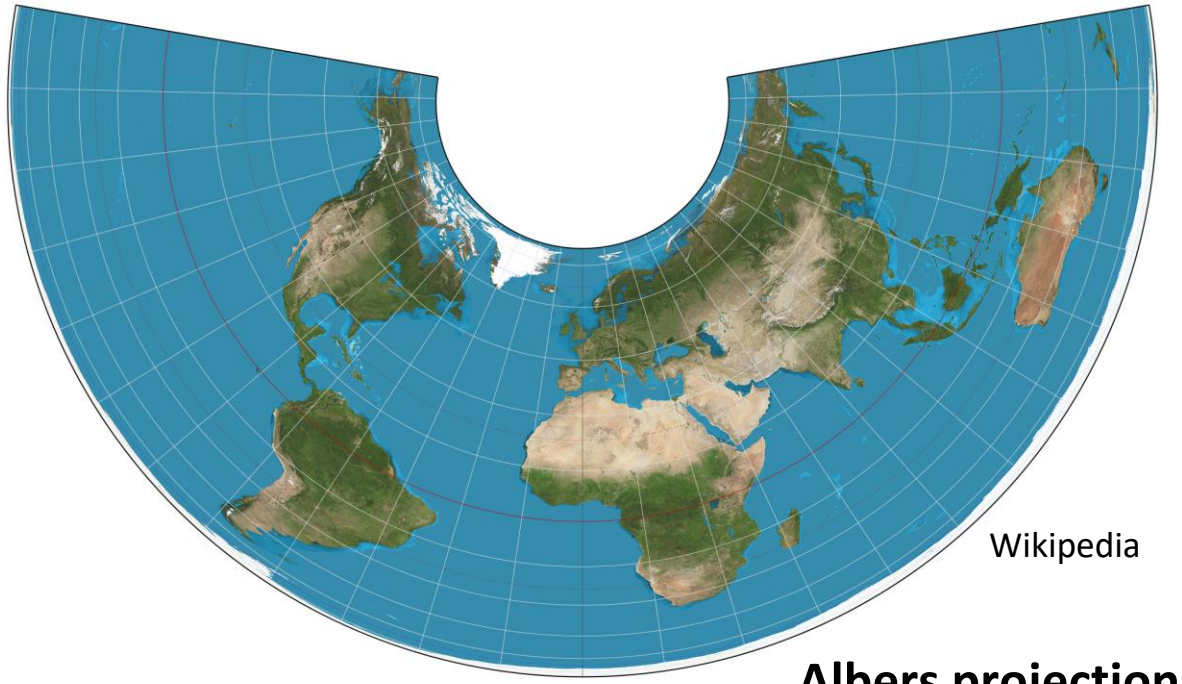
Projections

What projection to use?

Projections distort large parts of the Earth's surface – **need to choose the right projection for your map**

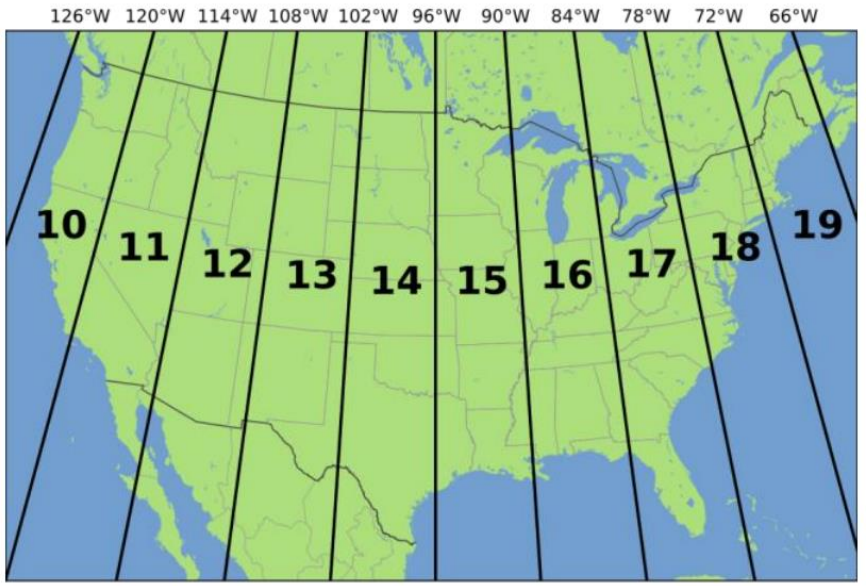
In BC, we want to use **BC Albers**
BC Albers: "equal area projection"

Northings and Eastings (metres/kilometres) instead of latitude/longitude (degrees)



Wikipedia

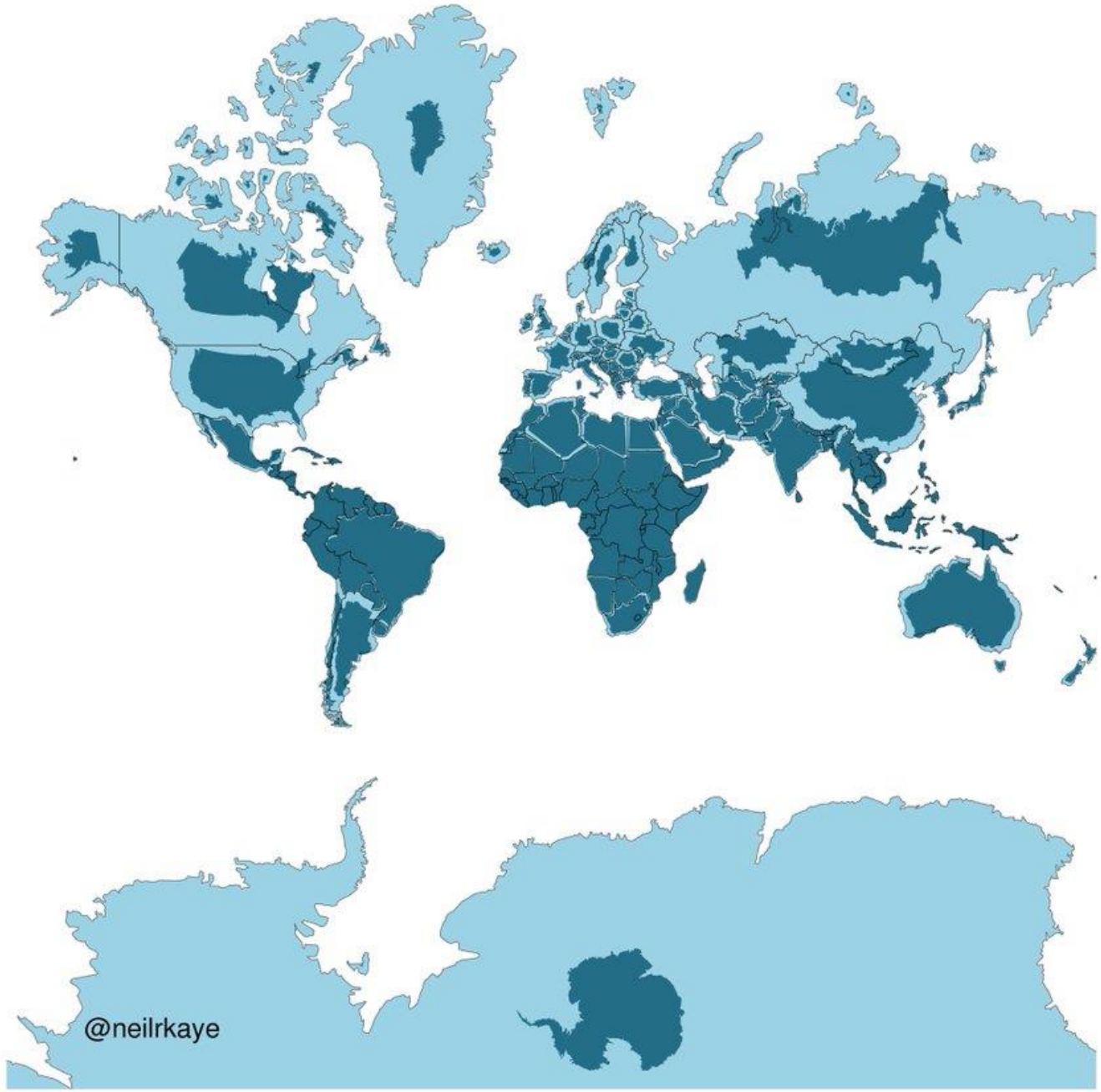
Albers projection



NCEAS

UTM zone 10

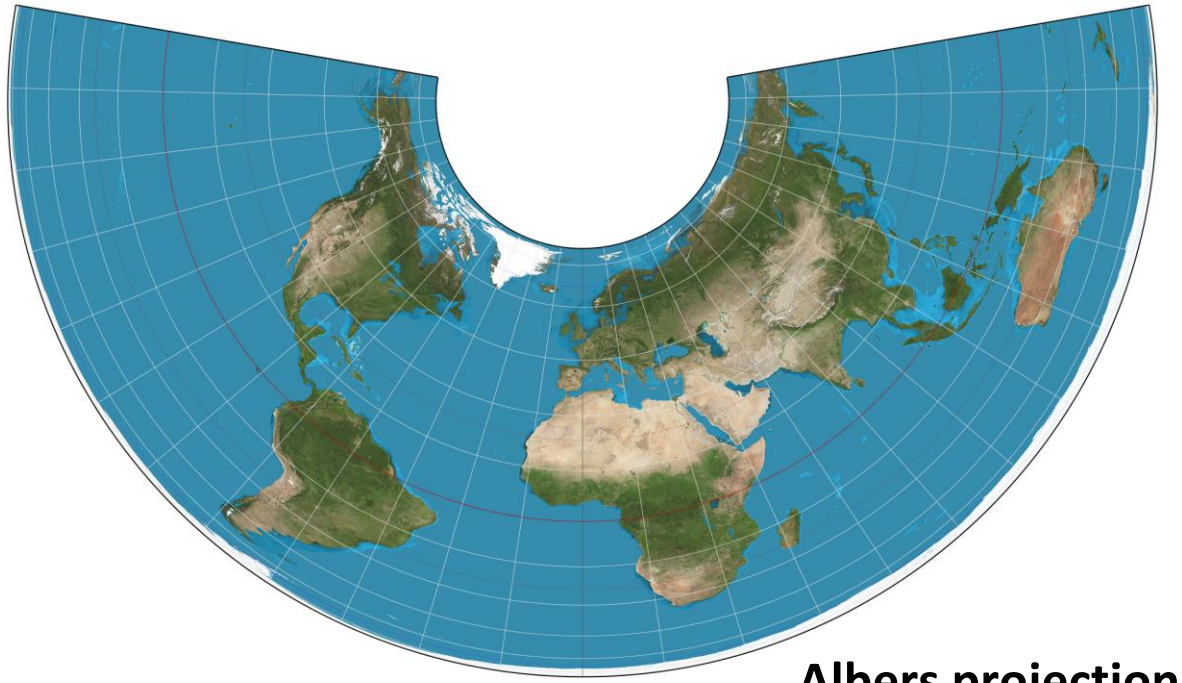
World Mercator projection with true country size and shape added



Projections

What projection to use?

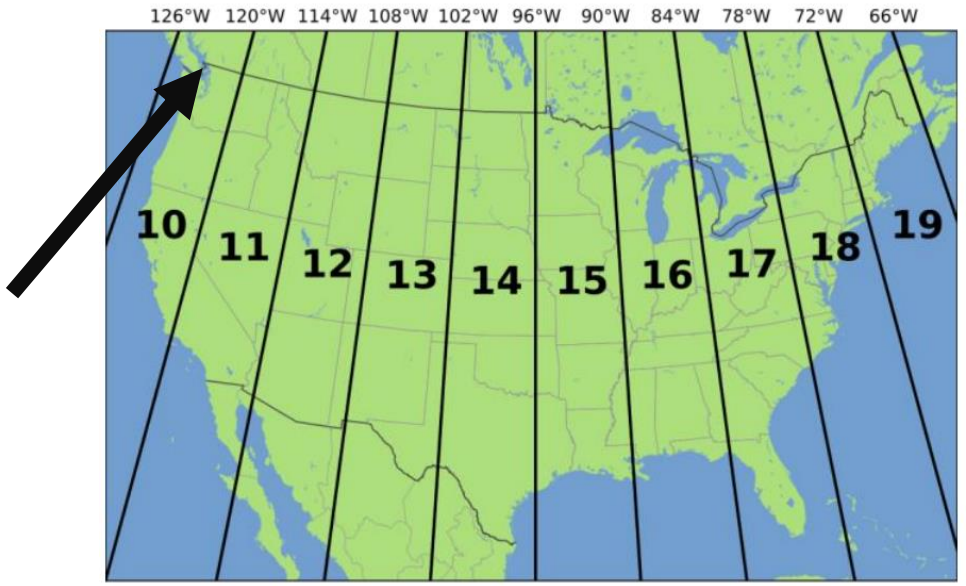
In this workshop, we will use WGS84 for simplicity



Albers projection

To transform to another projection:

```
albers <- st_transform(wgs), crs = st_crs(3005)
```

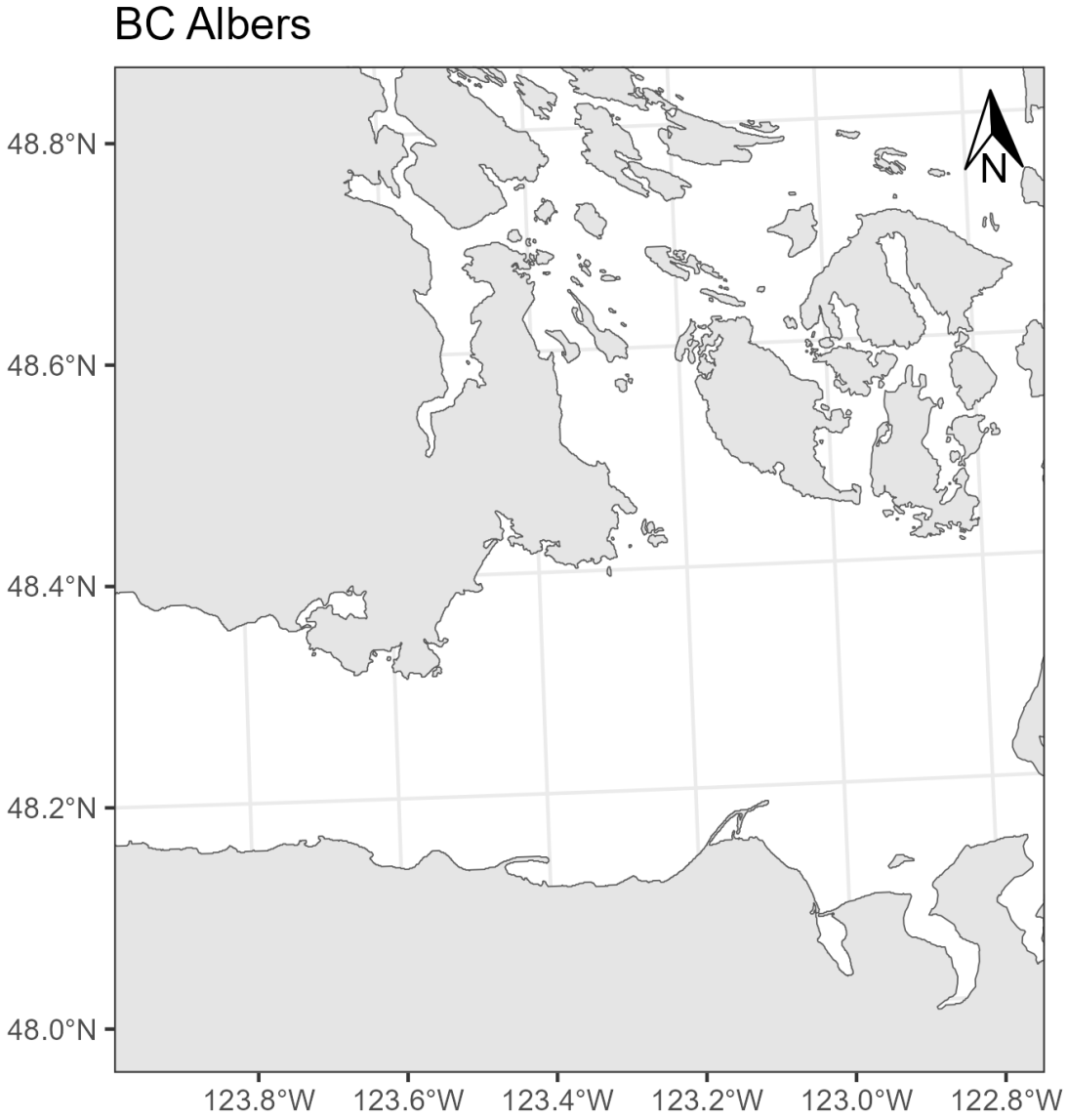


UTM zone 10

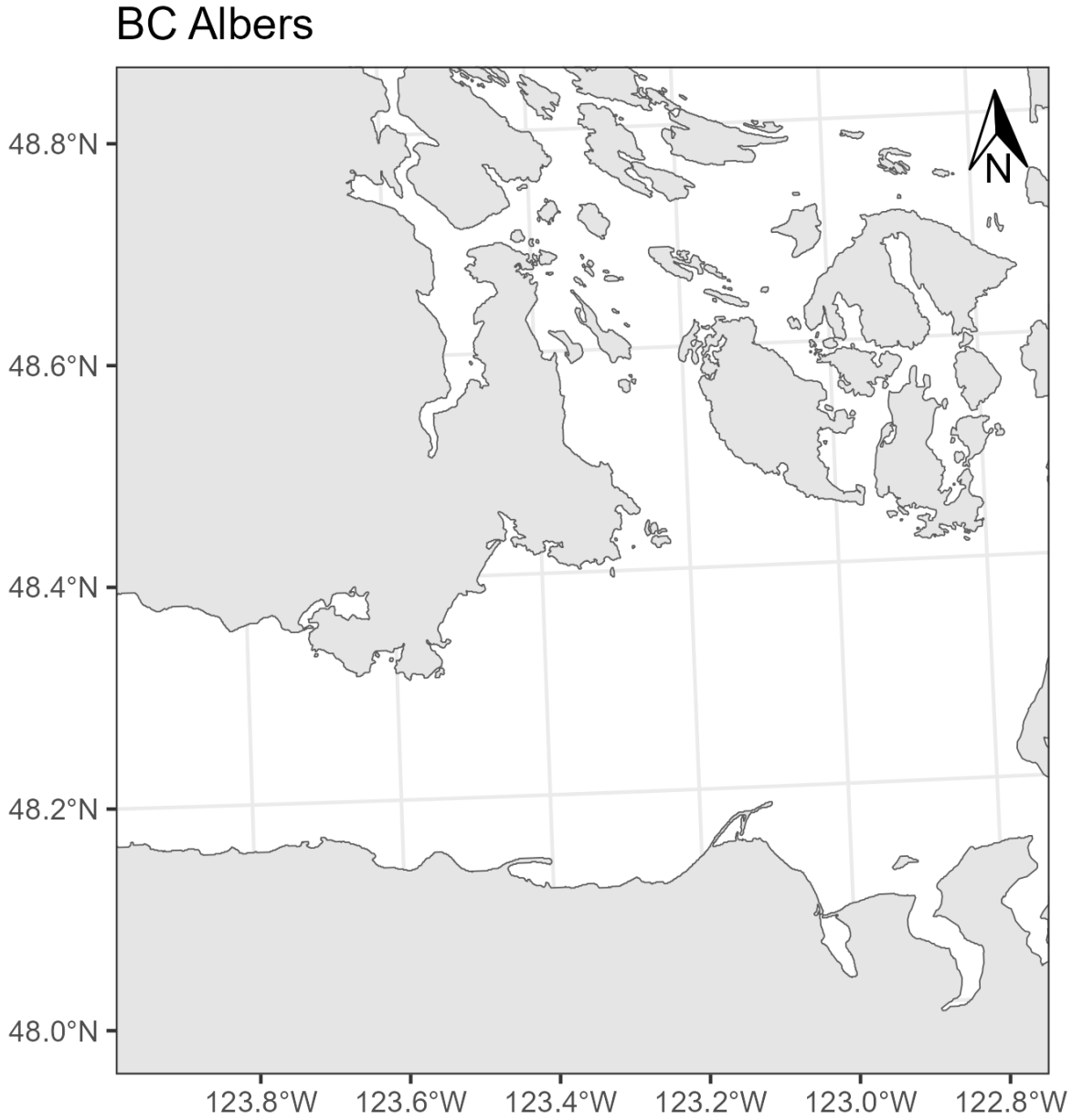
Projections

```
albers <- st_transform(wgs), crs = st_crs(3005)
```

See 00-study-area-map.R, section 4 for easy code for setting the bounding box of maps in a projection that uses metres or kilometres

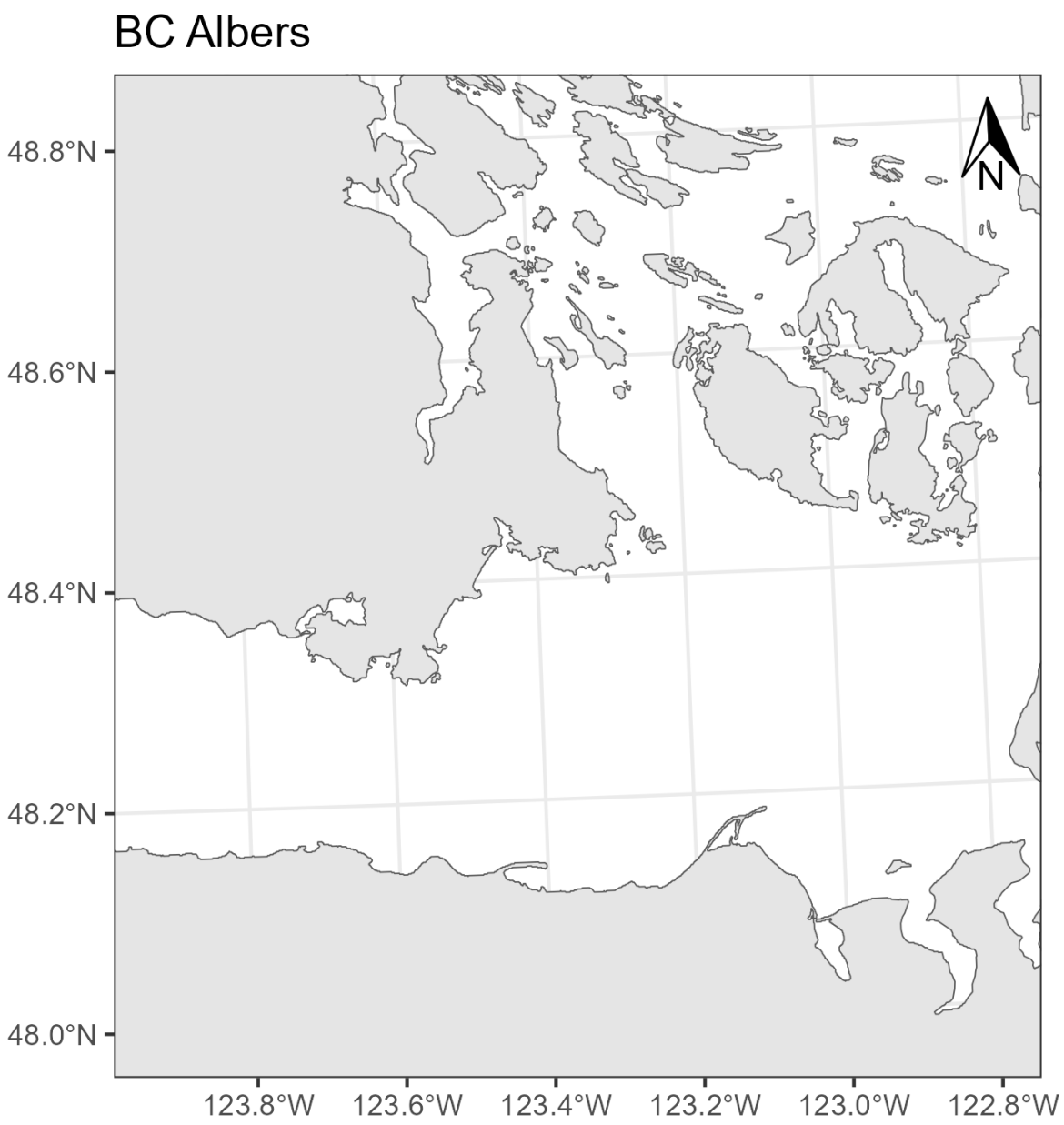


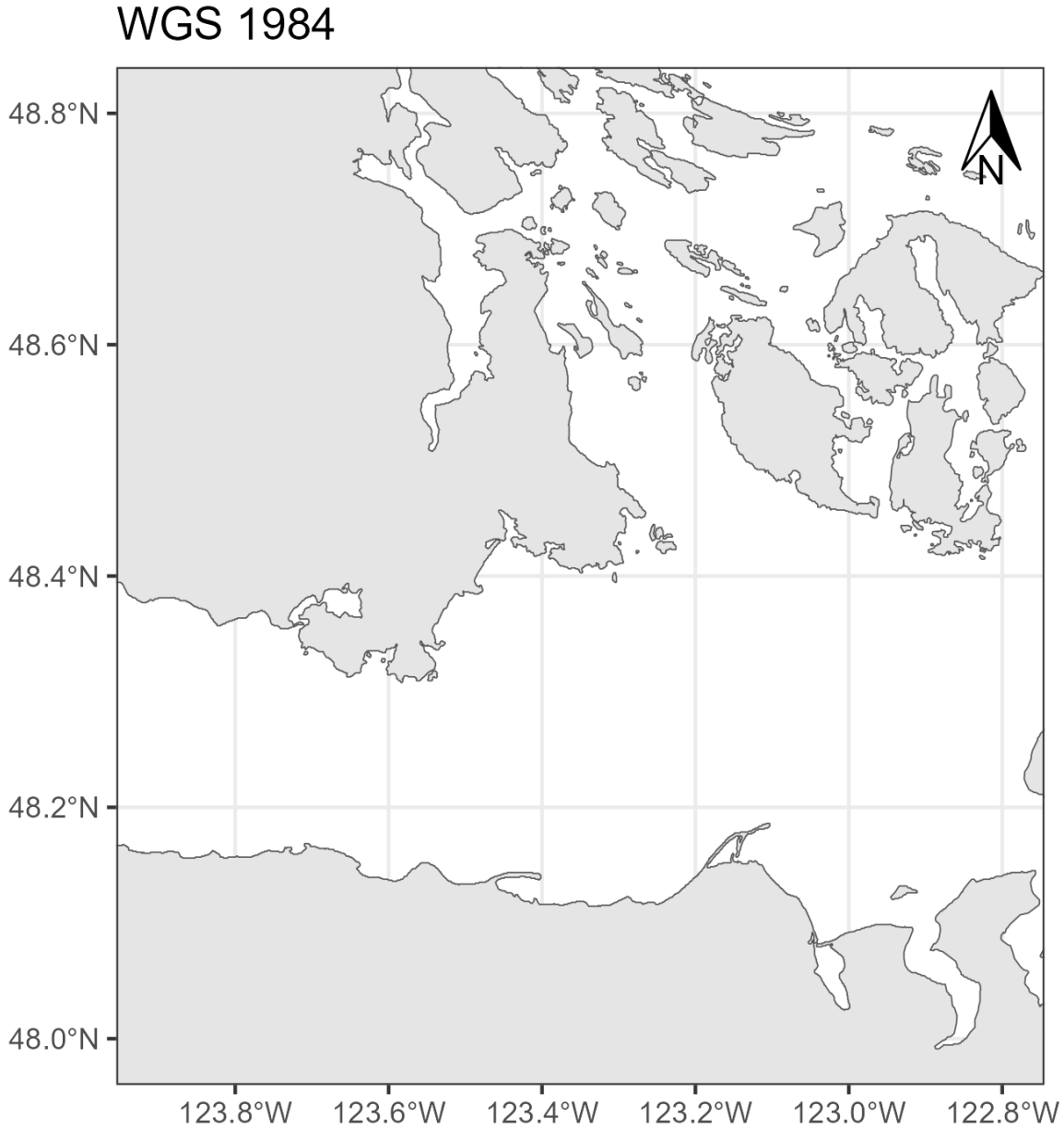
BC Albers study area map



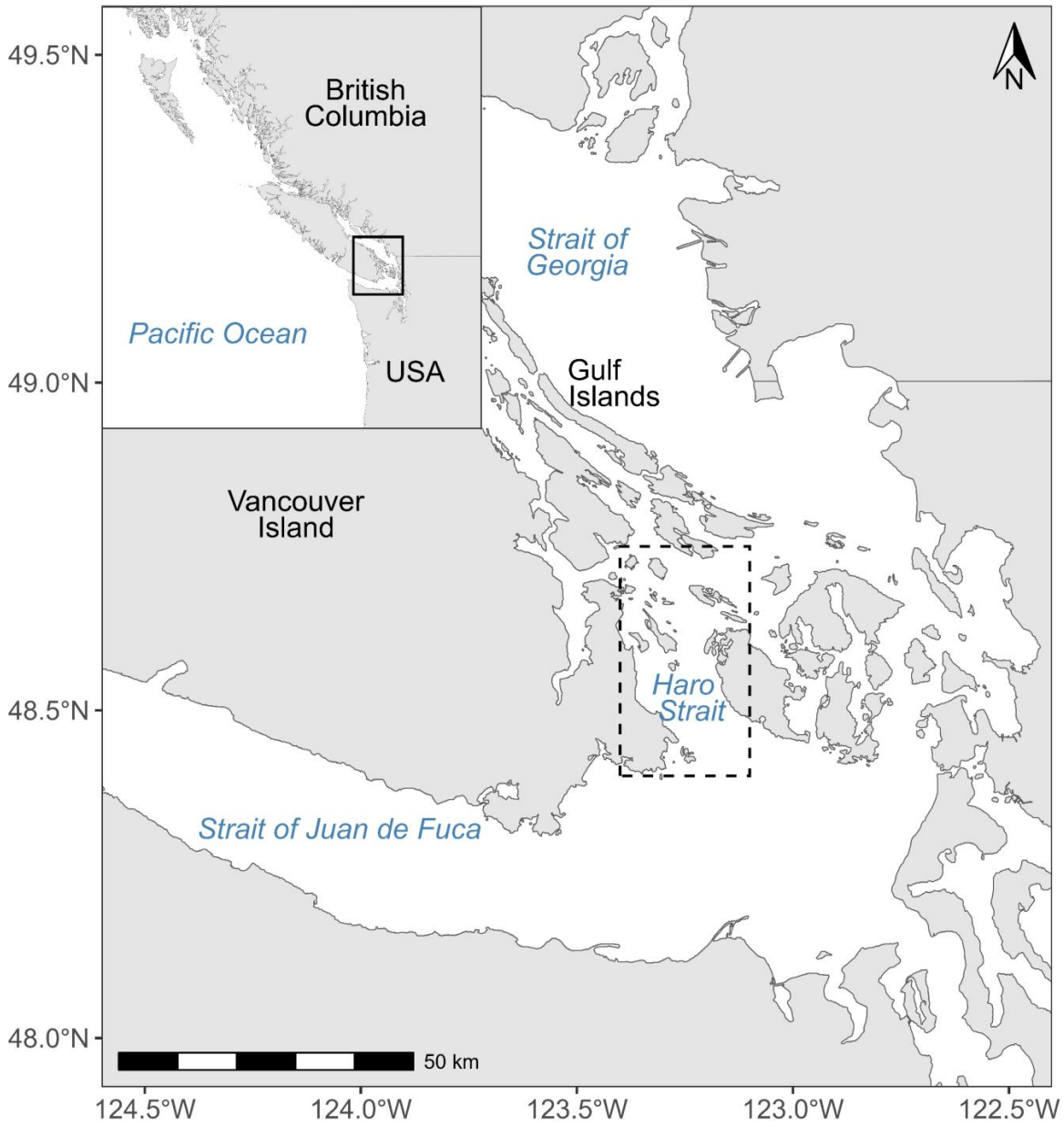
North arrow points slightly to the left

BC Albers study area map – slightly different





Simple study area map



Load basemap

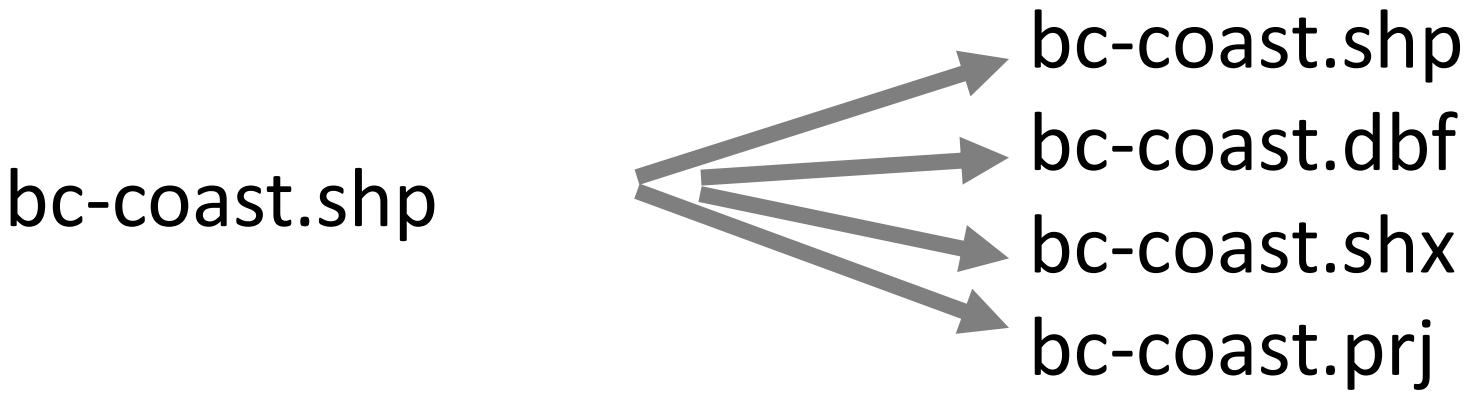
```
library(ggplot2)
library(sf)

bc.coast <- read_sf(“data/bc-coast.shp”)
```

Load basemap

```
bc.coast <- read_sf(“data/bc-coast.shp”)
```

A **shapefile** contains vector spatial data (e.g., points, lines, polygons).

A shapefile is NOT just one file!



When sharing shapefiles, remember you need all 4 files!

What projection?

```
library(ggplot2)
library(sf)

bc.coast <- read_sf("data/bc-coast.shp")
st_crs("data/bc-coast.shp")
```

st_crs() – what projection is the shapefile in?

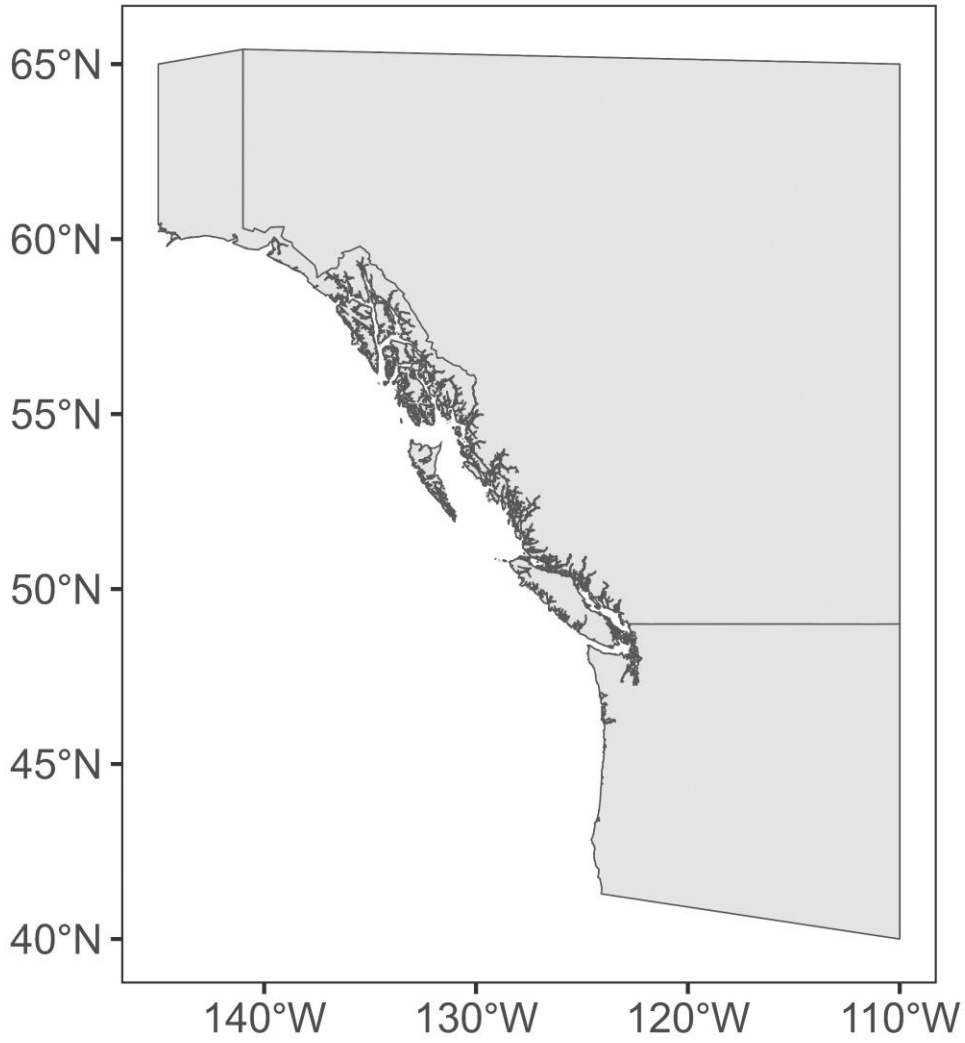
```
> st_crs(bc.coast)
Coordinate Reference System:
  User input: WGS 84
  wkt:
GEOGCRS["WGS 84",
    DATUM["World Geodetic System 1984",
        ELLIPSOID["WGS 84",6378137,298.257223563,
            LENGTHUNIT["metre",1]]],
    PRIMEM["Greenwich",0,
        ANGLEUNIT["degree",0.0174532925199433]],
    CS[ellipsoidal,2],
        AXIS["latitude",north,
            ORDER[1],
            ANGLEUNIT["degree",0.0174532925199433]],
        AXIS["longitude",east,
            ORDER[2],
            ANGLEUNIT["degree",0.0174532925199433]],
    ID["EPSG",4326]]
```

EPSG 4326: WGS 1984, a common global projection

We use EPSG numbers to refer to different projections/CRS
CRS: coordinate reference system

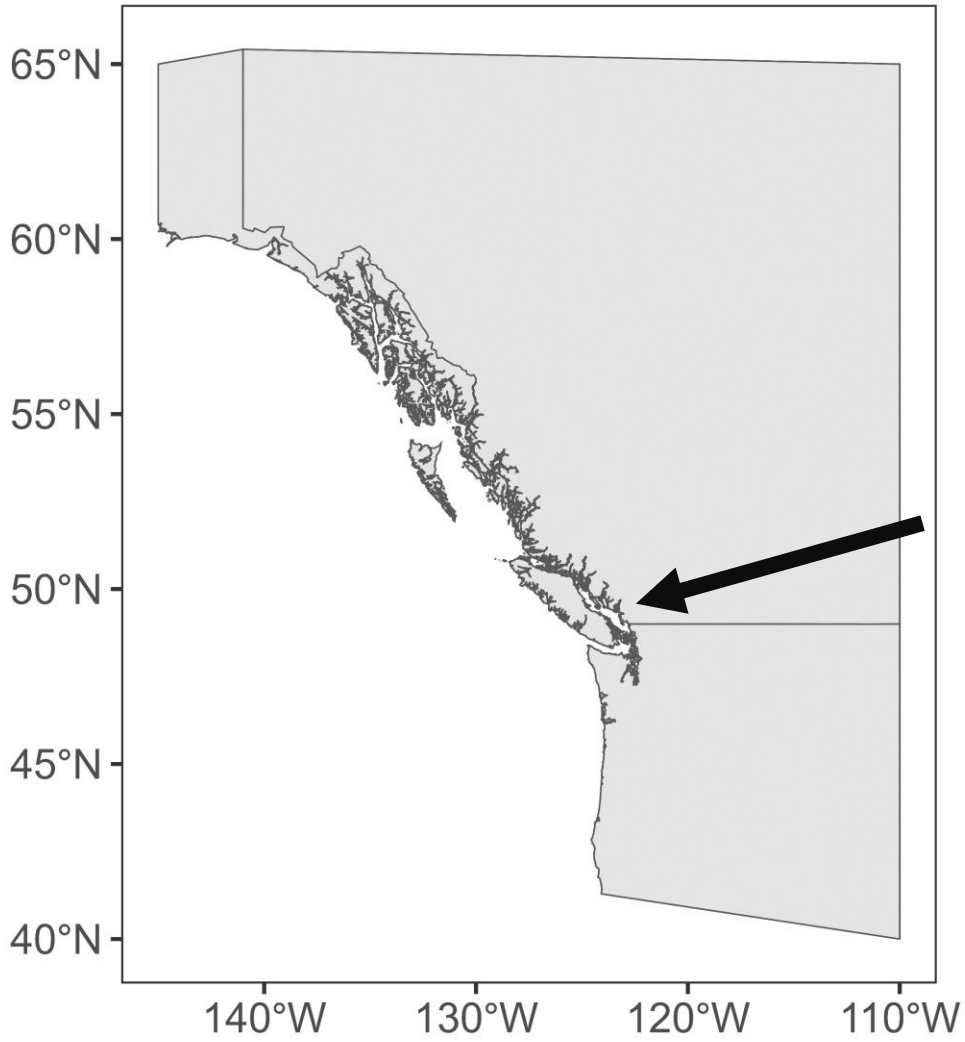
Plot basemap

```
ggplot() +
  geom_sf(data = bc.coast)
```



Plot basemap

```
ggplot() +
  geom_sf(data = bc.coast)
```

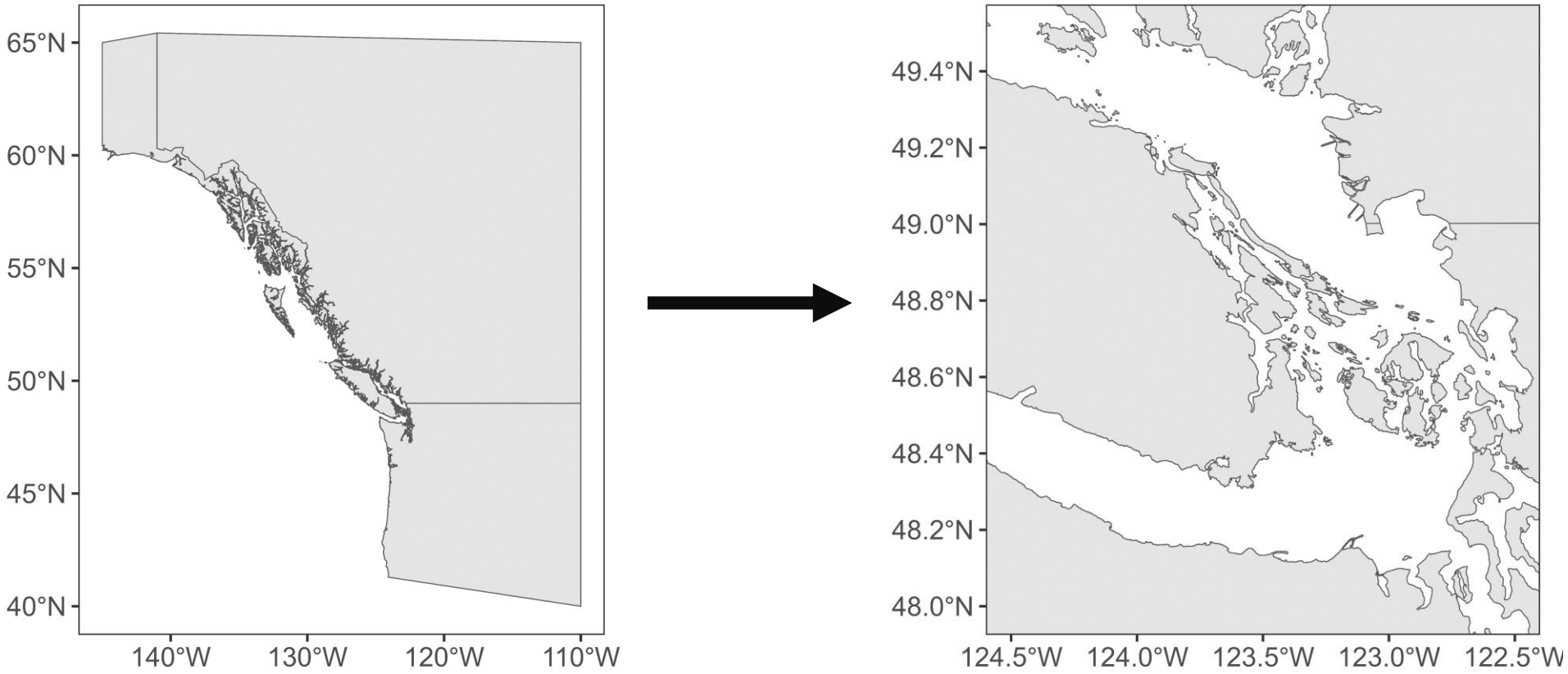


Need to zoom in on area of interest

Plot basemap and crop to area of interest

```
ggplot() +
  geom_sf(data = bc.coast) +

  coord_sf(xlim = c(-124.5, -122.5),
           ylim = c(48, 49.5))
```

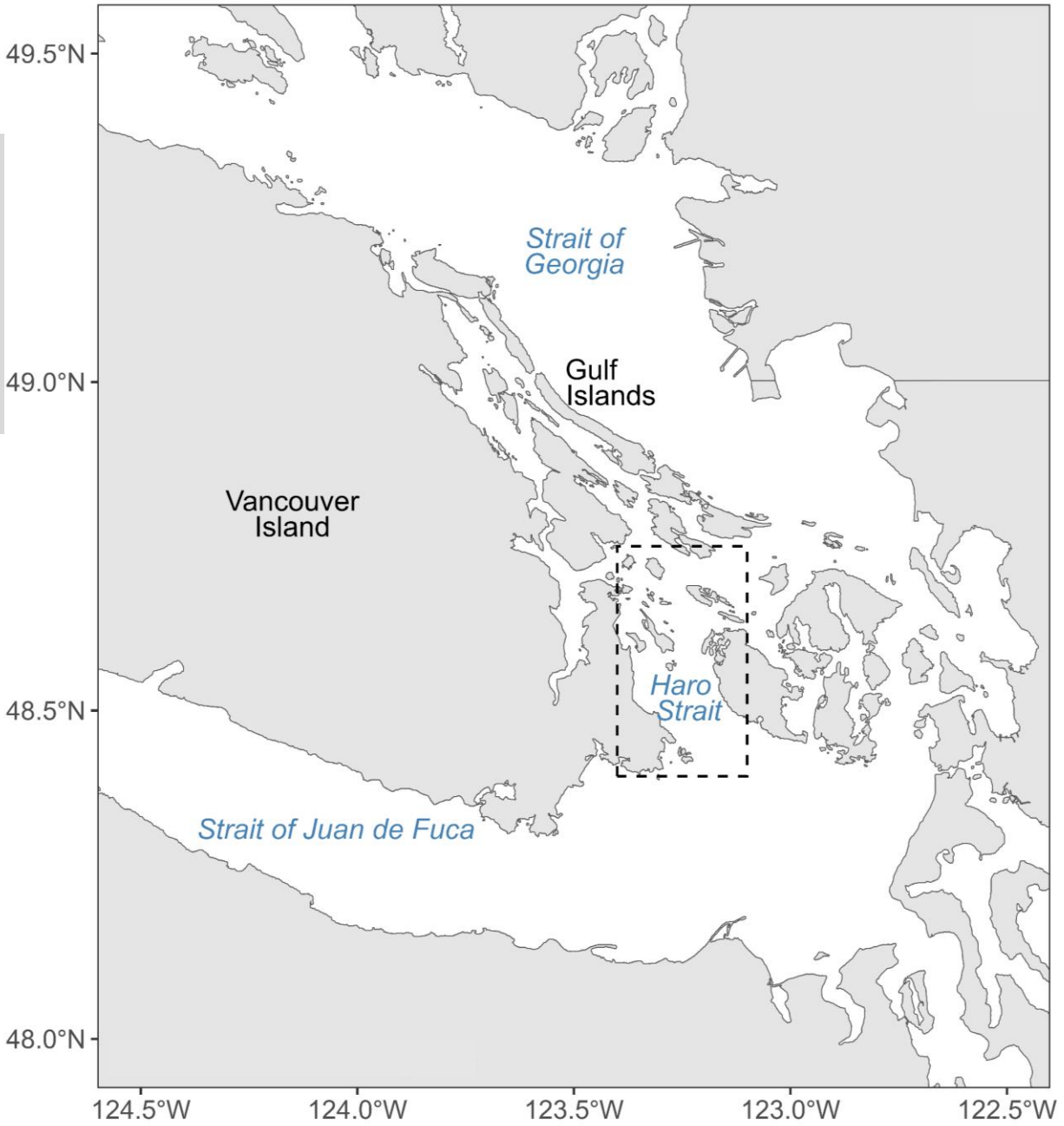
Crop bc.coast to our area of interest



Label areas of interest

```
# add scalebar (ggspatial package)

annotate(geom = “text”,
         x = -124, y = 48.3,
         label = “Strait of Juan de Fuca”)
```

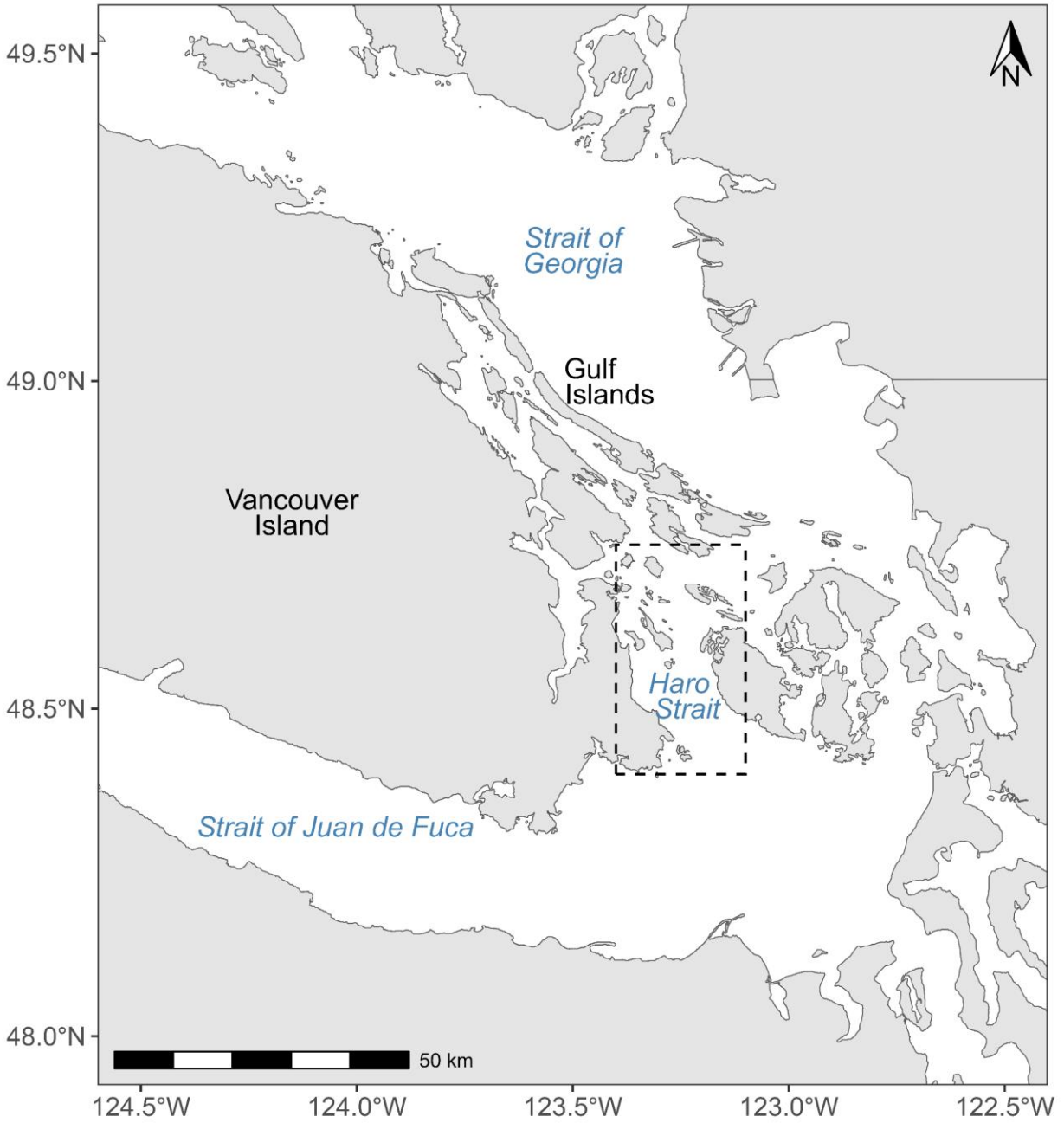
Personal preferences
Water bodies: *blue italics*
Land features: black



Add scalebar, north arrow

```
# add scalebar (ggspatial package)

annotation_scale(width_hint = 0.35) +

annotation_north_arrow(location = “tr”)
```

Note: the north arrow will point straight up for some projections but not others



Add inset map

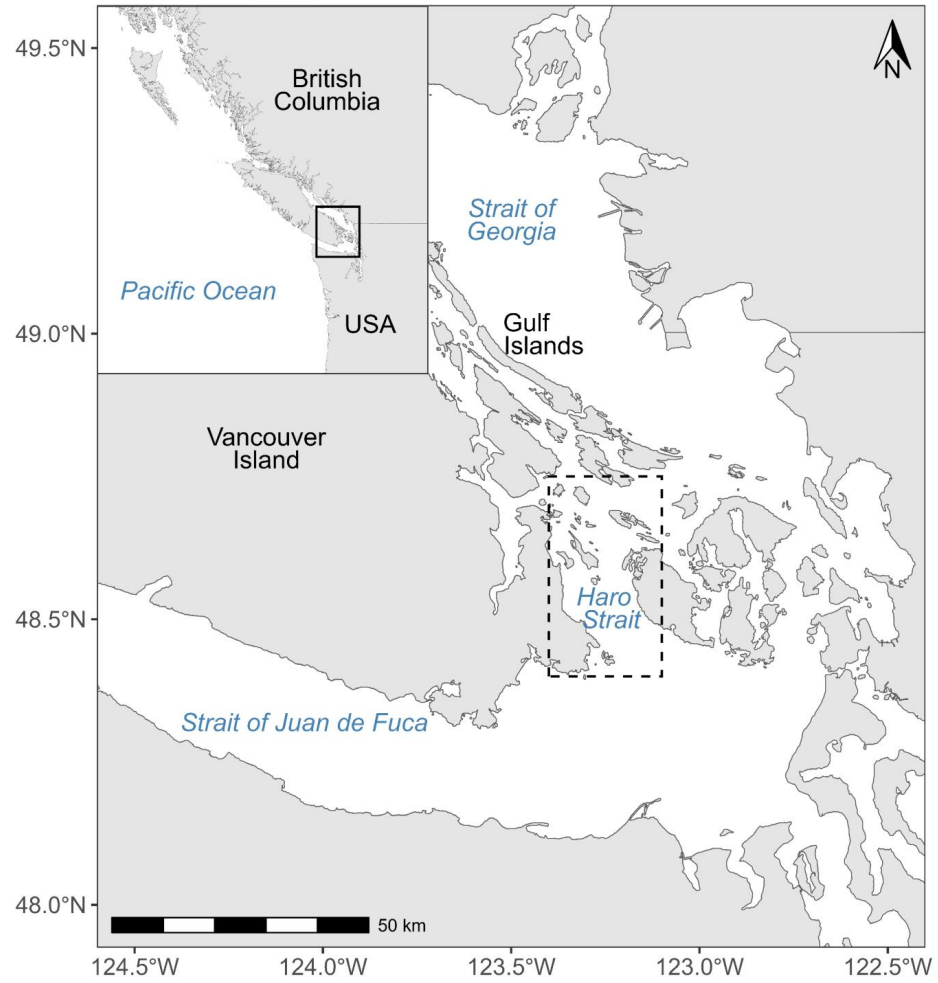
```
# inset map
inset <- ggplot() +
      geom_sf(data = bc.coast) +

      annotate(geom = “rect”,
                  xmin, xmax, ymin, ymax)
```

Add rectangle indicating the area of the larger map

Same coordinates as the coord_sf() call from the main map

Add inset map

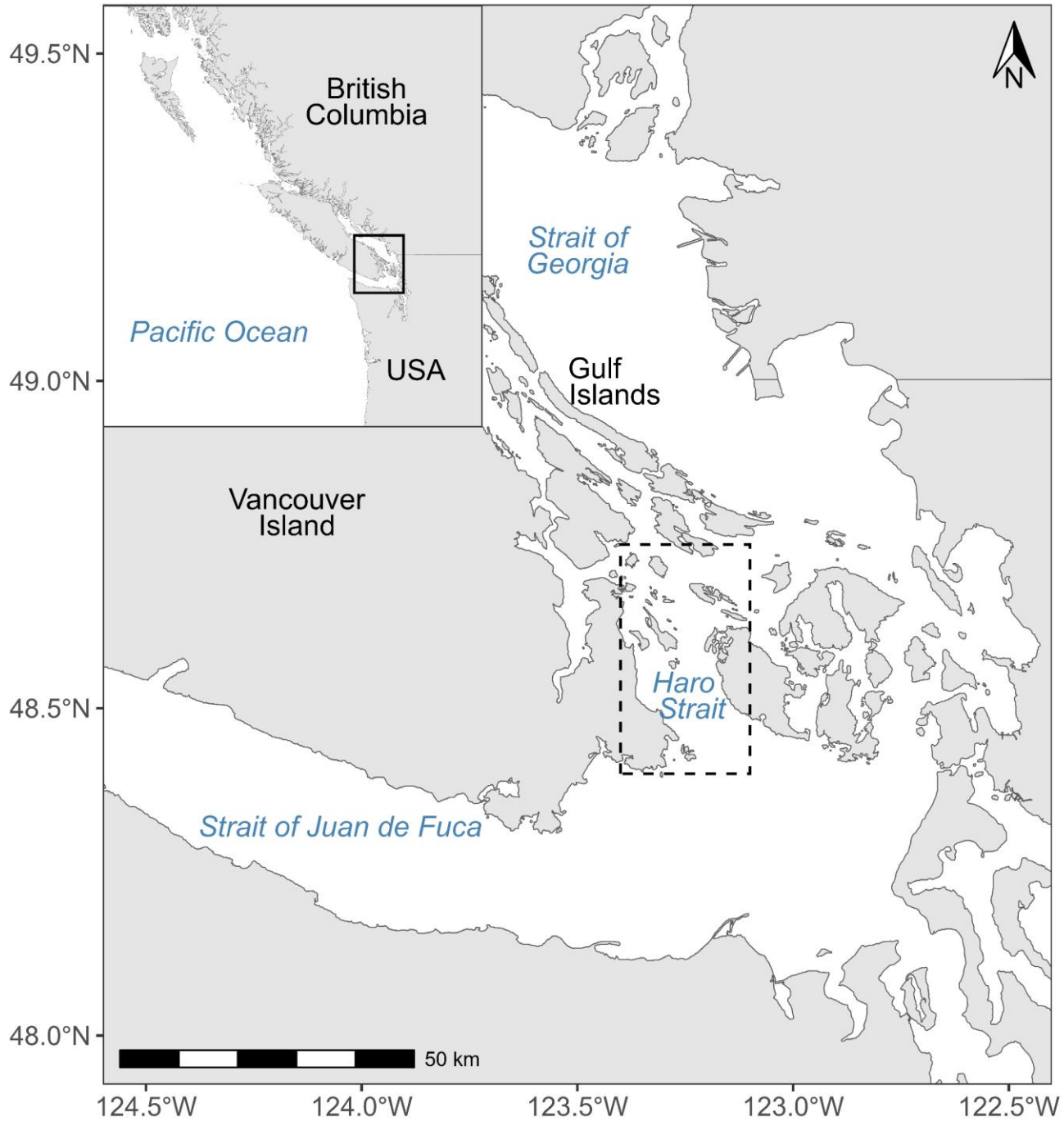
```
# add inset map

study.area +
      inset_element(inset,
      left = -0.007, bottom = 0.6,
      right = 0.4, top = 1.008,
      align_to = "panel")
```



left, right, bottom, top: Determine by trial and error so that inset map is desired size and location

Top-left inset, start with:
left = 0, bottom = 0.6, right = 0.4, top = 1

Top-right inset, start with:
left = 0, bottom = 0.6, right = 0.6, top = 1

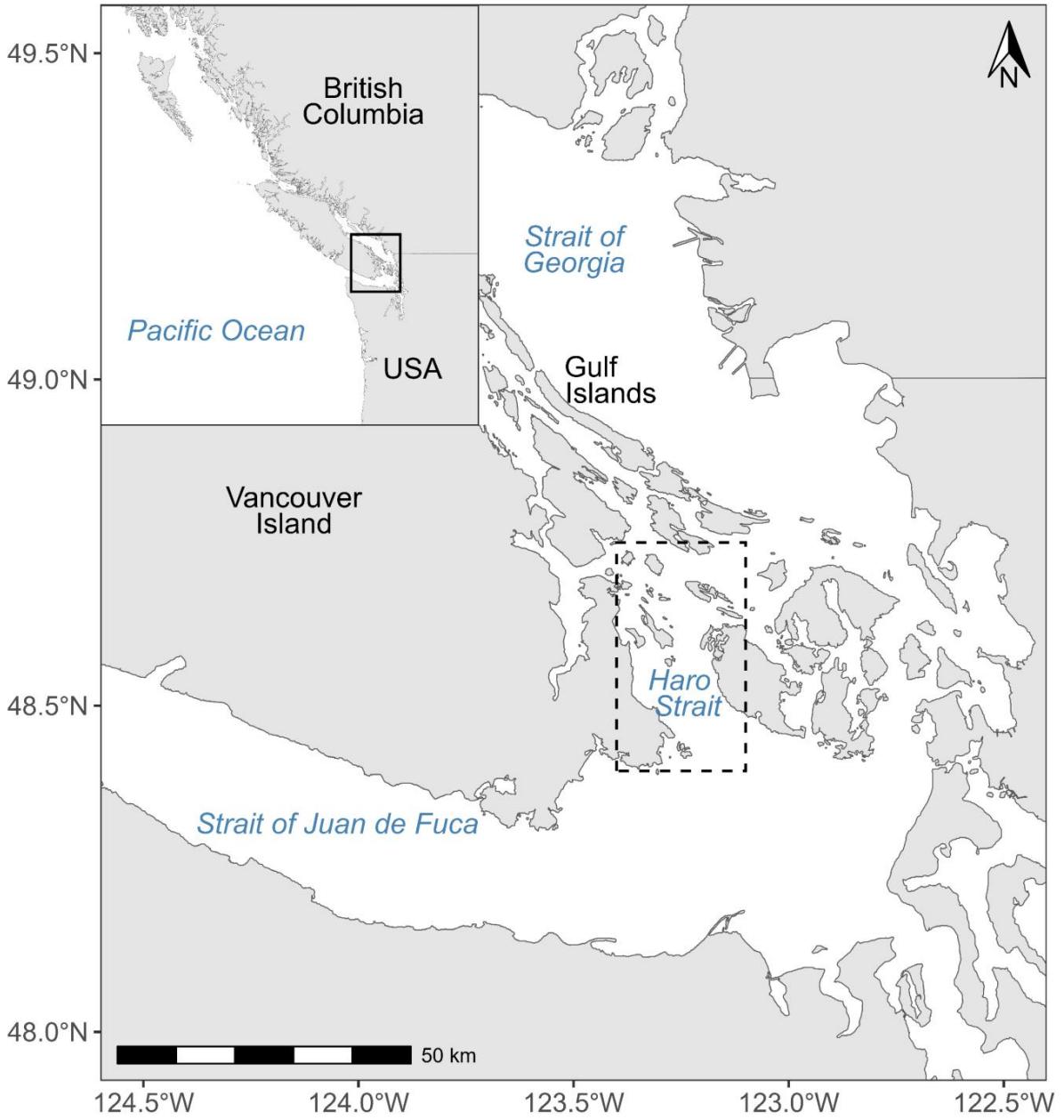

49.5°N
49.0°N
48.5°N
48.0°N
124.5°W
124.0°W
123.5°W
123.0°W
122.5°W
N
British Columbia
Pacific Ocean
USA
Strait of Georgia
Gulf Islands
Vancouver Island
Haro Strait
Strait of Juan de Fuca
50 km

Save the map

Print the map, then use ggsave()

```
study.area +
      inset_element(inset,
      left = -0.007, bottom = 0.6,
      right = 0.4, top = 1.008,
      align_to = "panel")

ggsave("figures/study-area-map.PNG",
       width = 17, height = 17,
       units = "cm", dpi = 300)
```

Can increase resolution to dpi = 1600 for very high quality images, if desired



Where to find additional data layers?

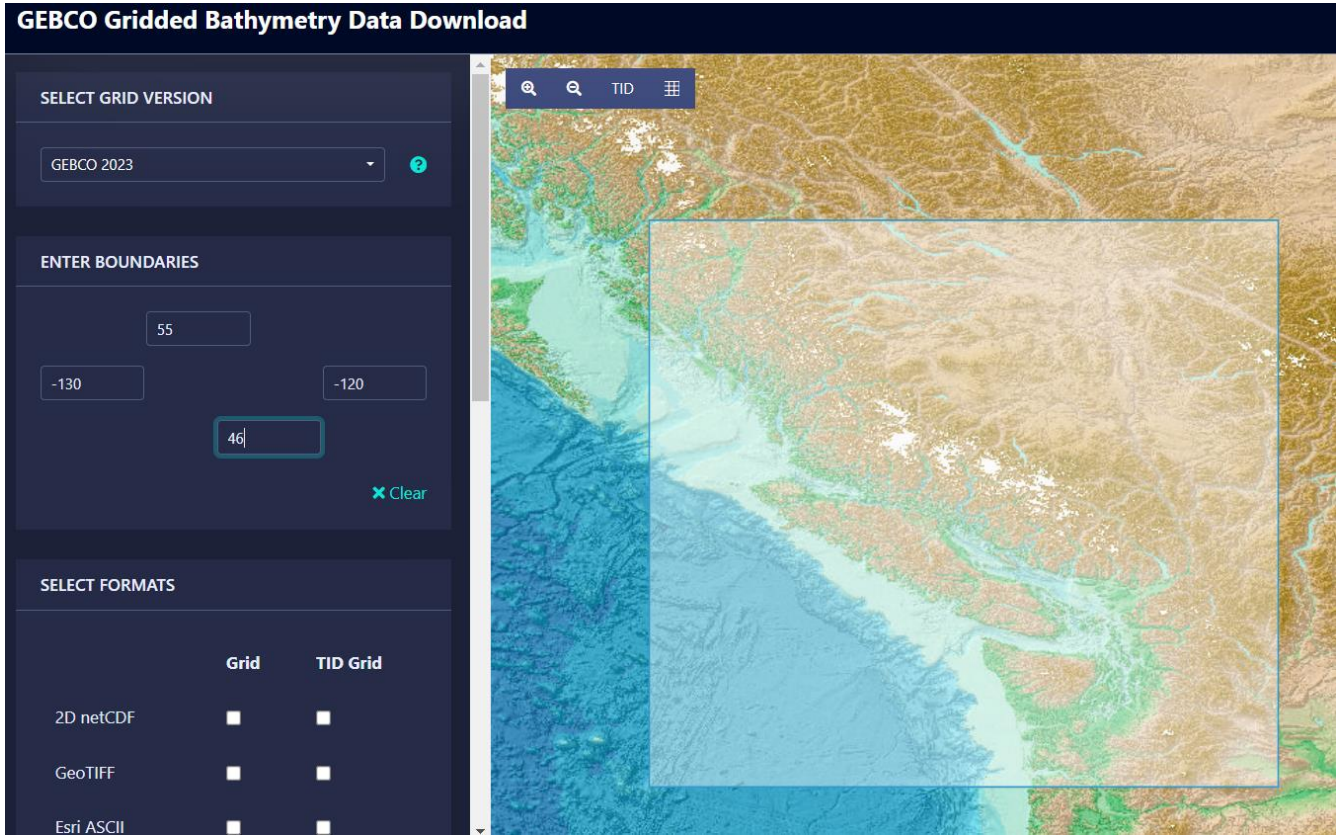
Lots of open-source spatial data available

- BC Government Data Catalogue (especially terrestrial, freshwater)
  https://catalogue.data.gov.bc.ca/
- PSF/UBC Strait of Georgia Data Centre and Marine Reference Guide
  https://soggy2.zoology.ubc.ca/geonetwork/srv/eng/catalog.search#/home
  https://gis.sogdatacentre.ca/sog-mrg/
- GEBCO (bathymetry and topography)
  https://www.gebco.net/

Add bathymetry and topography

- Using digital elevation model downloaded from GEBCO
- **Raster** of cells with a z-value indicating height below or above sea level

- **`terra` package: raster data**



https://download.gebco.net/

GEBCO Gridded Bathymetry Data Download

SELECT GRID VERSION

GEBCO 2023

ENTER BOUNDARIES

55

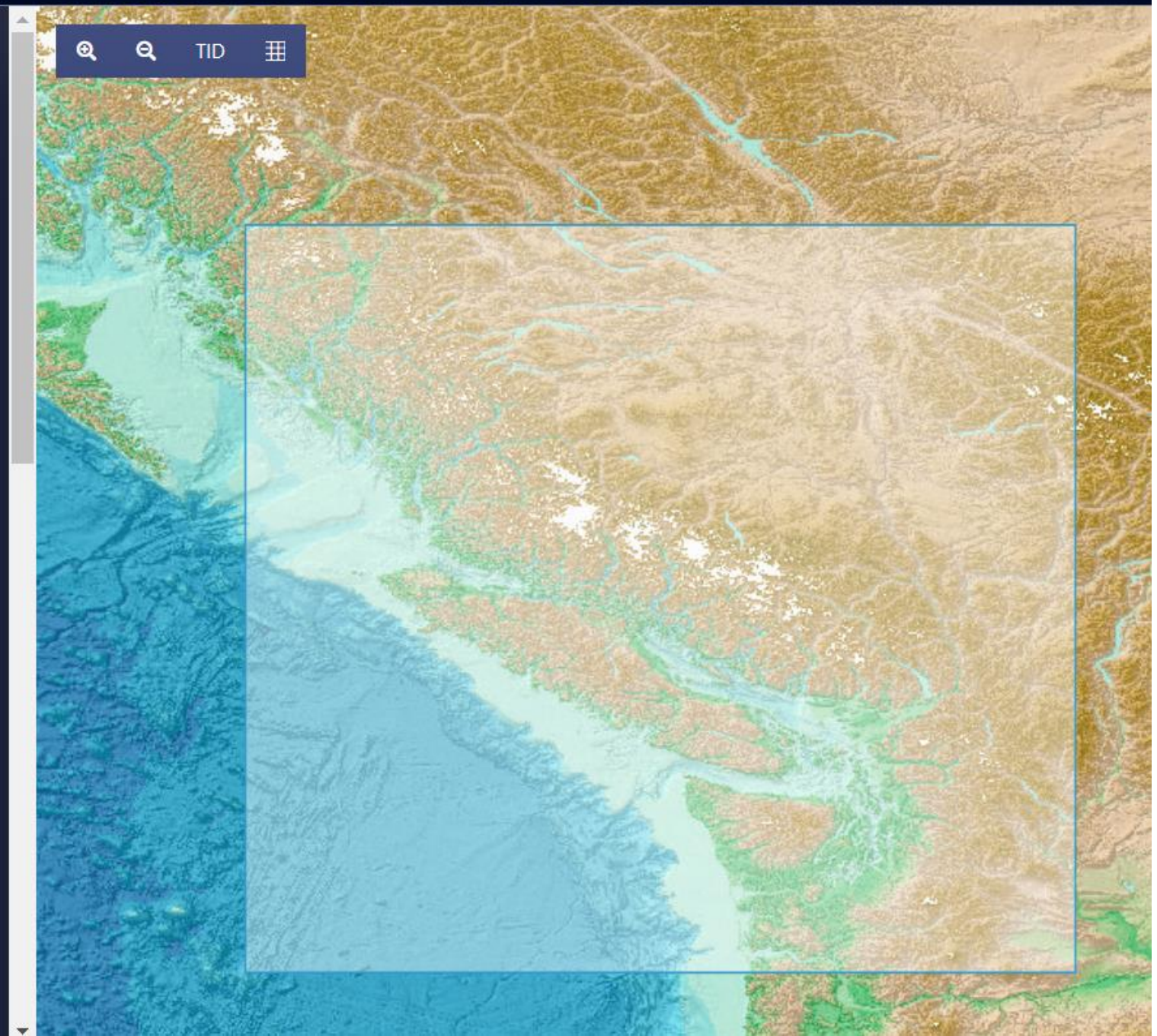
-130 -120

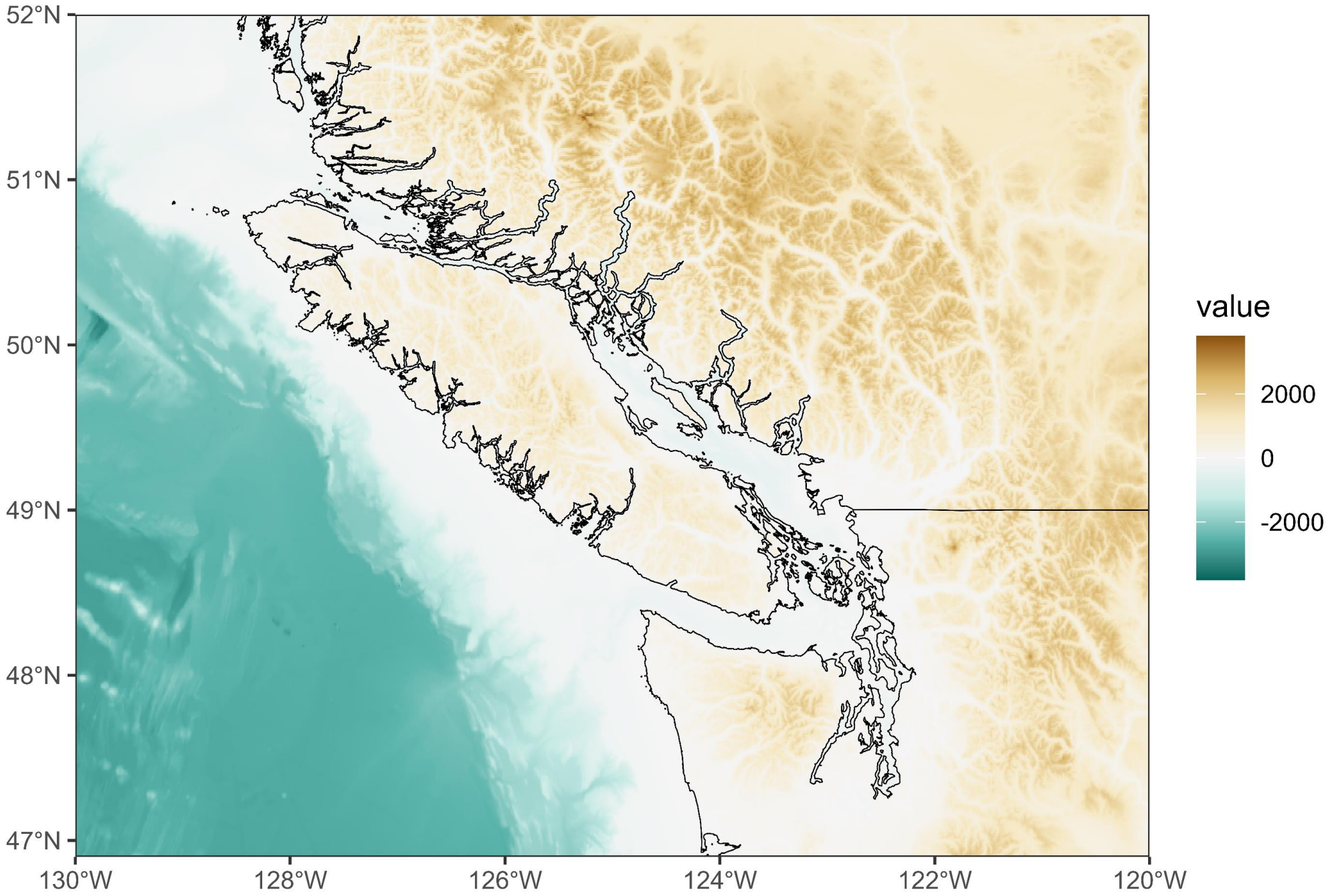
46

Clear

SELECT FORMATS

	Grid	TID Grid
2D netCDF	☐	☐
GeoTIFF	☐	☐
Esri ASCII	☐	☐




52°N
51°N
50°N
49°N
48°N
47°N
130°W
128°W
126°W
124°W
122°W
120°W
value
2000
0
-2000

Add bathymetry and topography

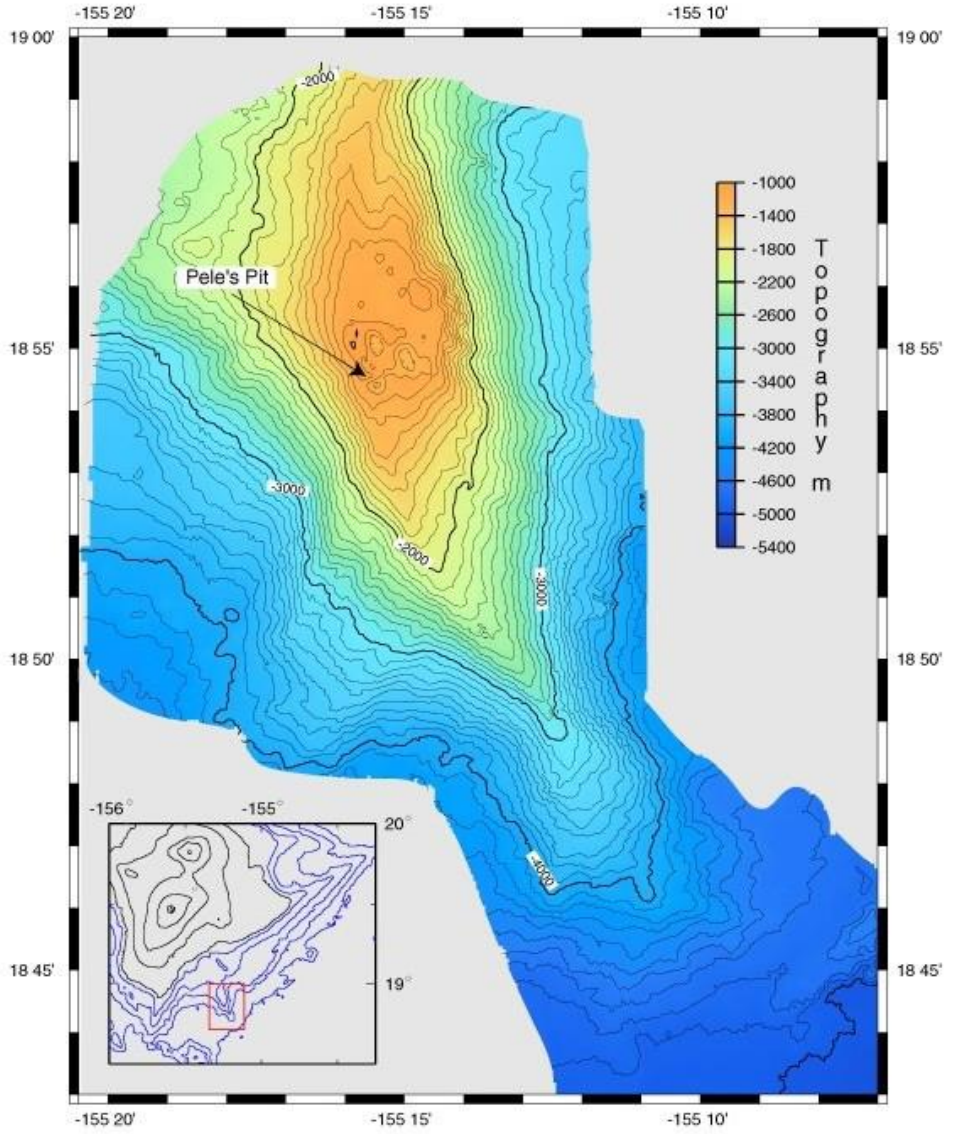
`terra` package: raster data

Plot rasters with
`geom_spatraster()` from
`tidyterra` package

Add bathymetry and topography

Bathymetry layer: depth of each grid cell

Topography layer: “hillshading” - the shadows cast by the digital elevation model, if the sun was shining from northwest at a 45° angle

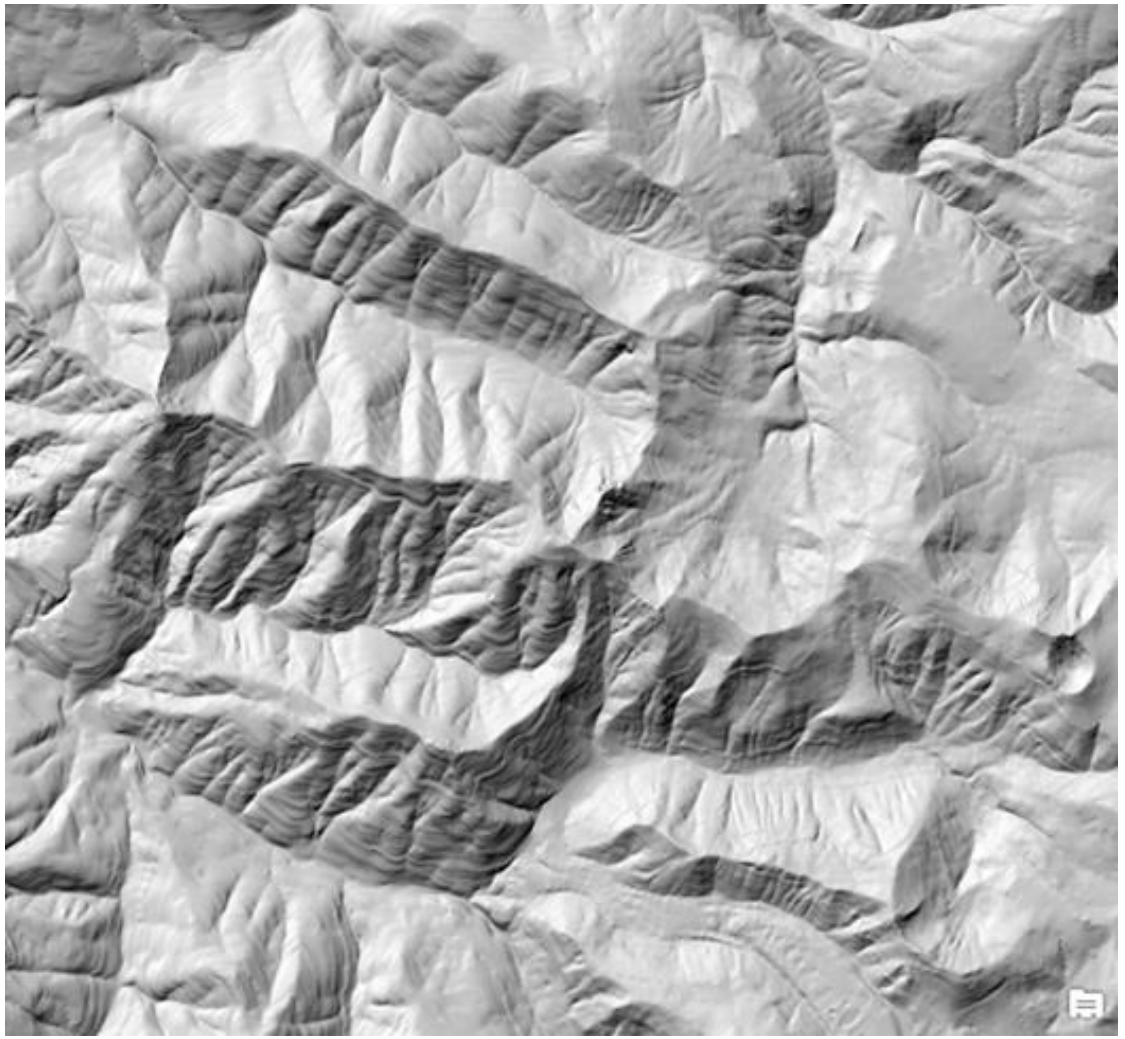


Wikipedia

Add bathymetry and topography

Bathymetry layer: depth of each grid cell, with land filtered out

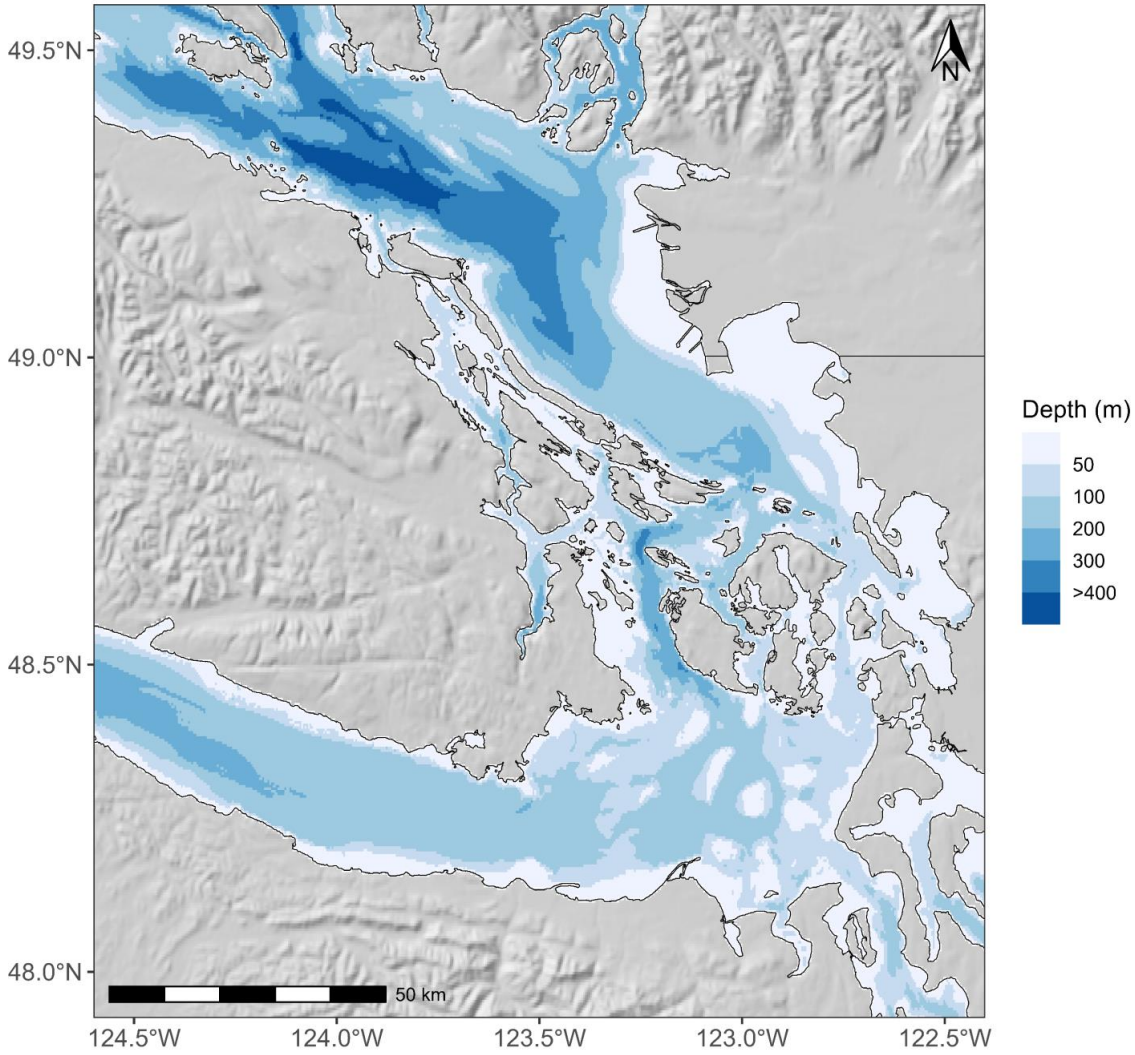
Topography layer: “hillshading” - the shadows cast by the digital elevation model, if the sun was shining from northwest at a 45° angle



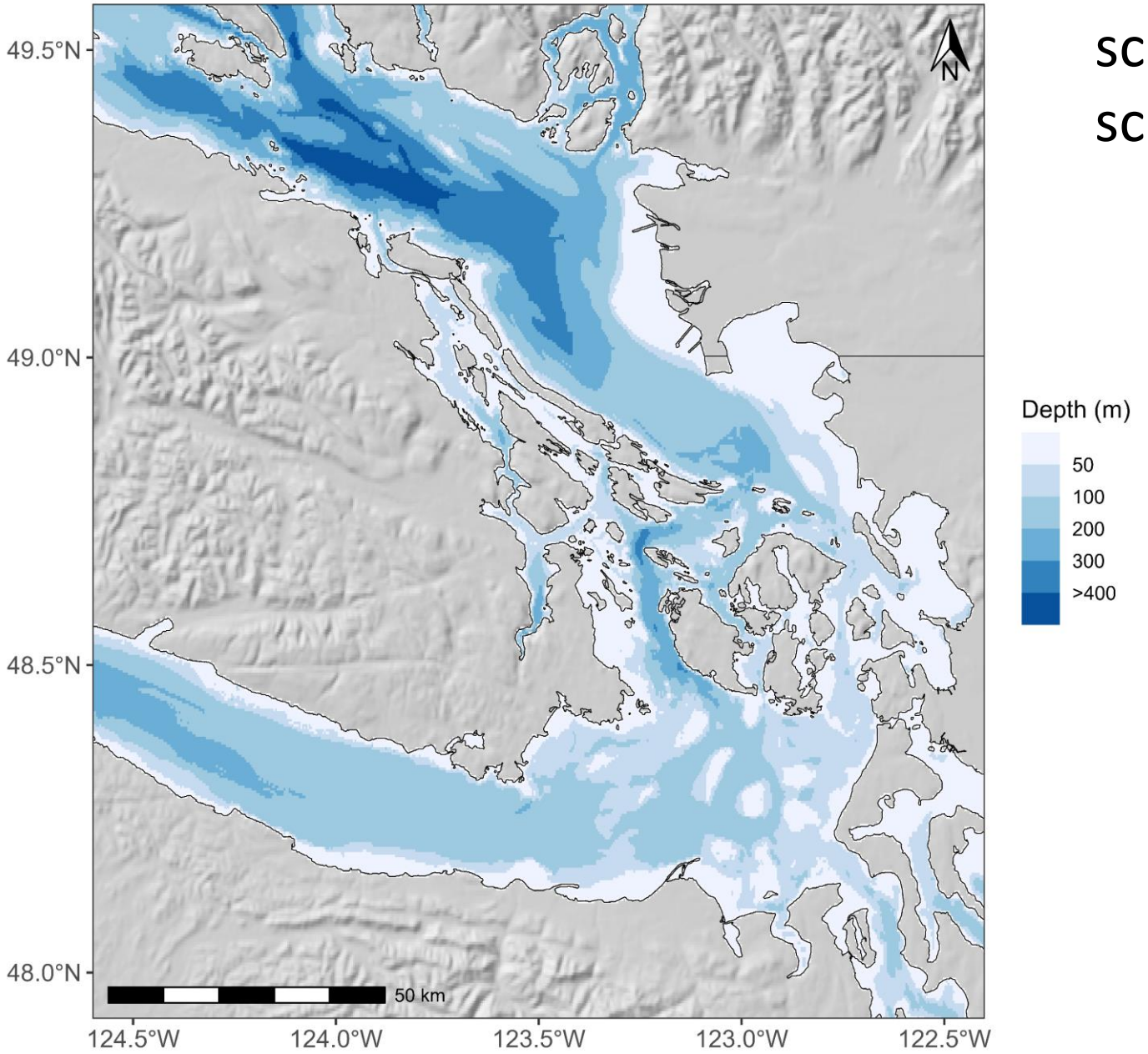
ArcGIS

Add bathymetry and topography

```
ggplot() +
       geom_spatraster(data = hillshading,
                       alpha = 0.7) +

       geom_spatraster(data = bathymetry) +

       geom_sf(bc.coast)
```



Add bathymetry and topography



Full code:
scripts/bathymetry.R
scripts/topography.R

Add external polygon and point data

Hakai

Rockfish conservation areas (RCAs): marine protected areas with restricted fisheries to protect rockfishes

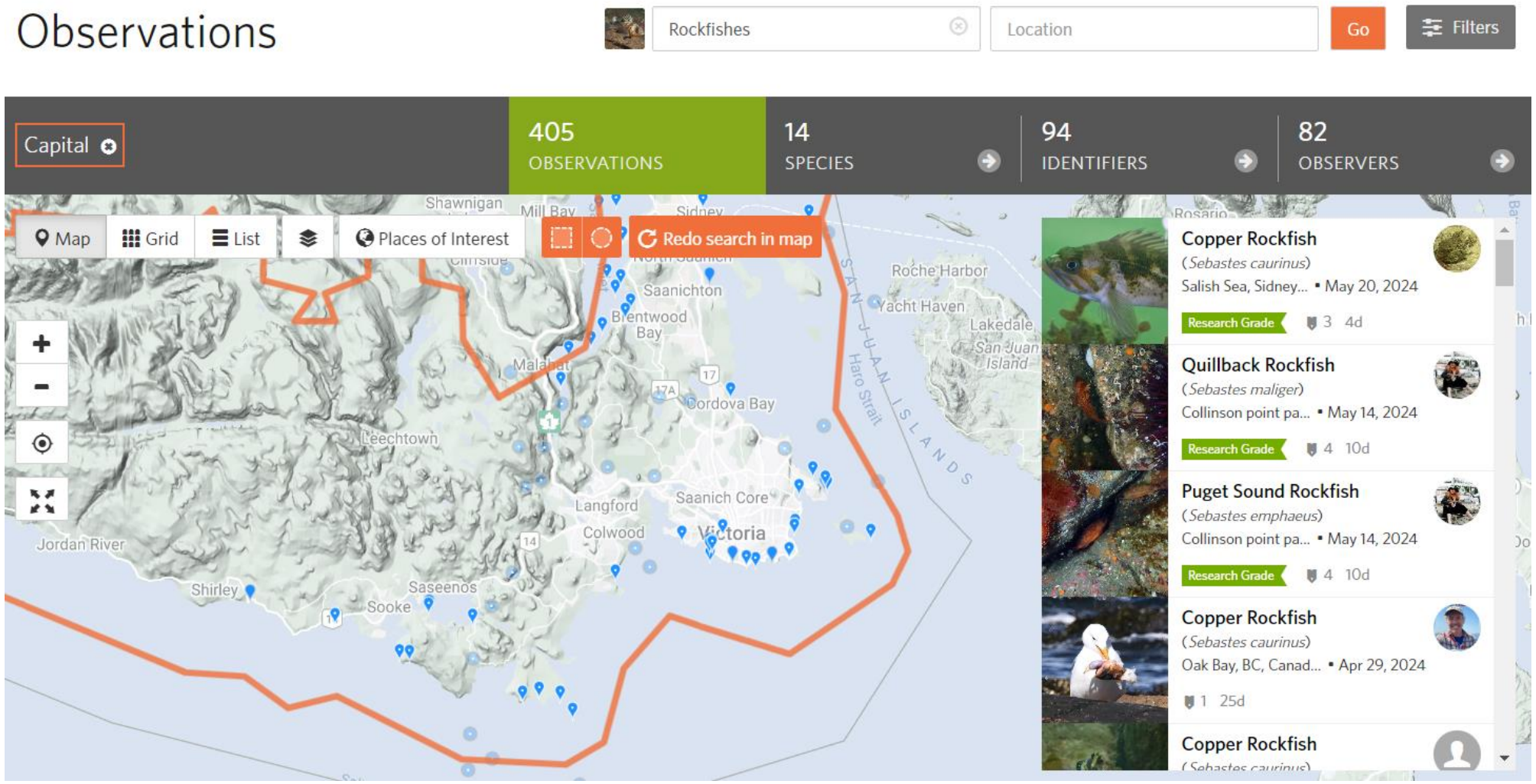
Add external polygon and point data

Hakai

Rockfish conservation areas (RCAs): marine protected areas with restricted fisheries to protect rockfishes

Question: What is the distribution of RCAs relative to rockfish observations on iNaturalist around Victoria?

iNaturalist data



https://www.inaturalist.org/observations?place_id=49140&subview=map&taxon_id=47762

Add rockfish conservation areas

What projection is the RCA shapefile?

```
rca <- read_sf("data/rca.sf")

st_crs(rca)
```

```
  .. .. ..@ projargs: chr "+proj=aea +lat_0=40 +lon_0=-96 +lat_1=50 +lat_2=70 +x_0=
0 +y_0=0 +datum=NAD83 +units=m +no_defs"
  .. .. ..$ comment: chr "PROJCRS[\"Canada_Albers_Equal_Area_Conic\",\n    BASEGEOG
CRS[\"NAD83\",\n        DATUM[\"North American Datum 1"| __truncated__
  ..$ comment: chr "TRUE"
>
```

Canada Albers Equal Area Conic

Add rockfish conservation areas

Transform RCA layer

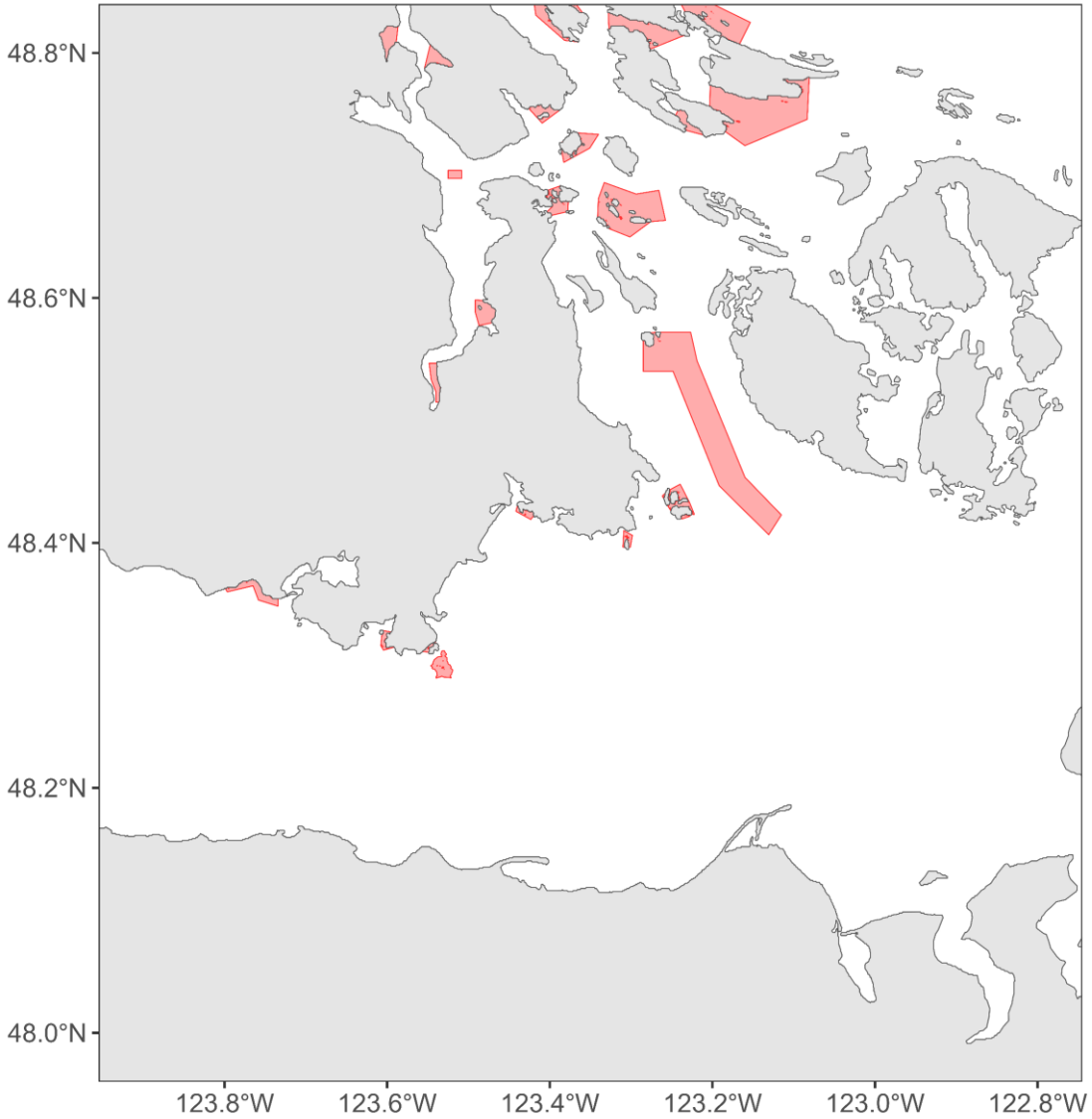
```
rca <- rgdal::readOGR("data/rca_shapefile", "rockfish_102001")

str(rca)
```

```
  .. .. ..@ projargs: chr "+proj=aea +lat_0=40 +lon_0=-96 +lat_1=50 +lat_2=70 +x_0=
0 +y_0=0 +datum=NAD83 +units=m +no_defs"
  .. .. ..$ comment: chr "PROJCRS[\"Canada_Albers_Equal_Area_Conic\",\n    BASEGEOG
CRS[\"NAD83\",\n        DATUM[\"North American Datum 1"| __truncated__
  ..$ comment: chr "TRUE"
>
```

Canada Albers Equal Area Conic

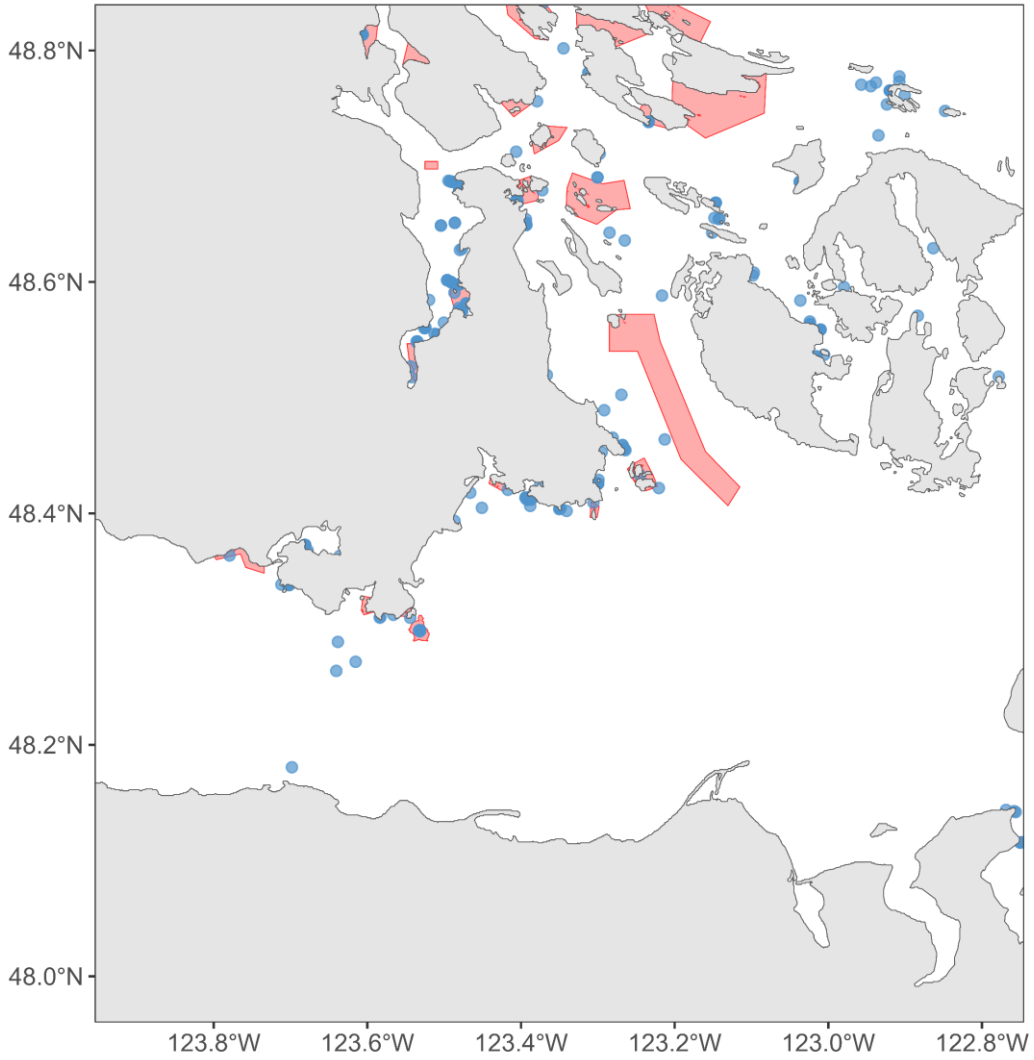
Add rockfish conservation areas

```
geom_sf(data = rca, fill = "firebrick1")
```

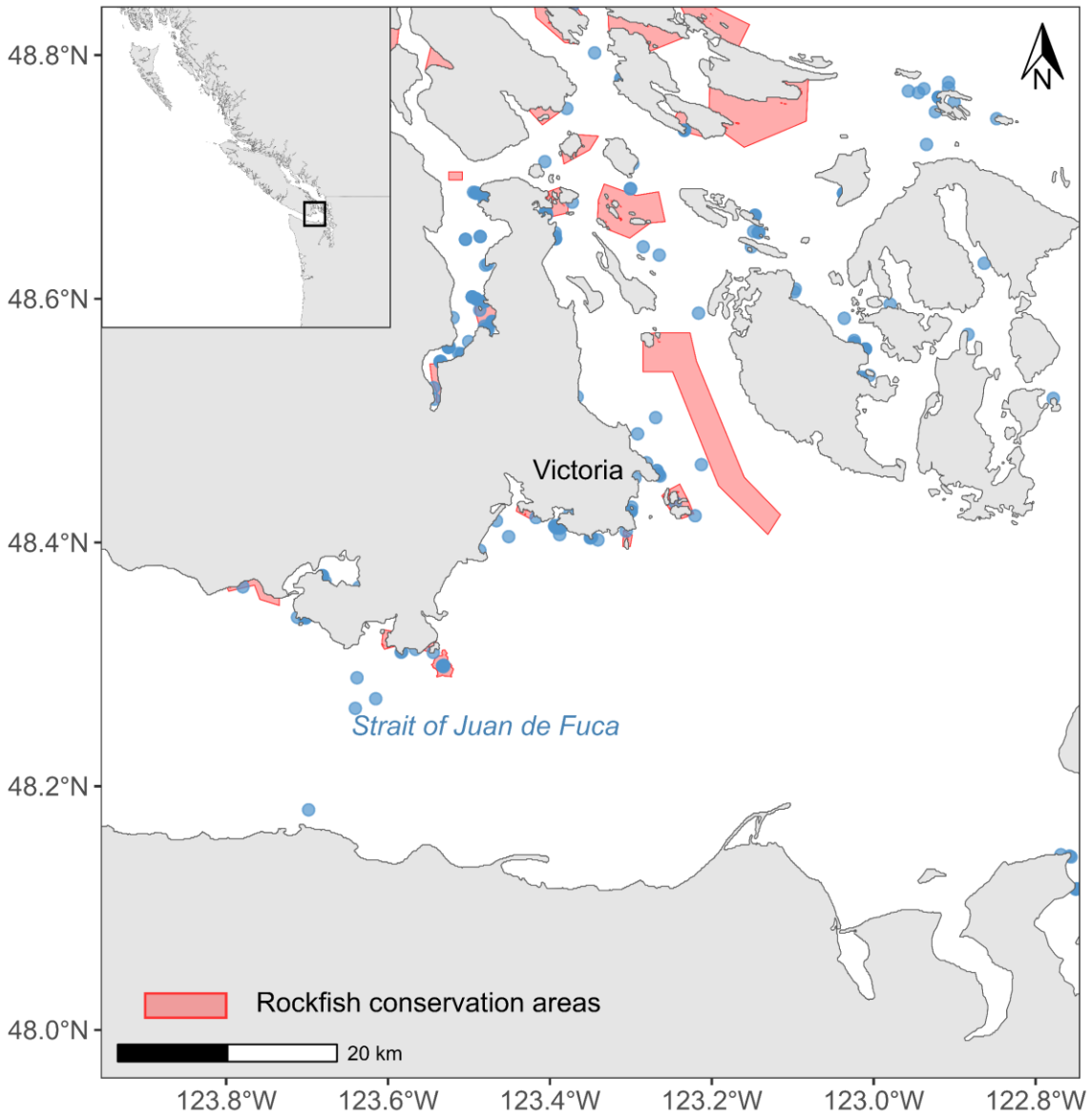


Add iNaturalist observations

```
geom_point(data = iNaturalist.rockfish,
           aes(x = longitude, y = latitude)
```

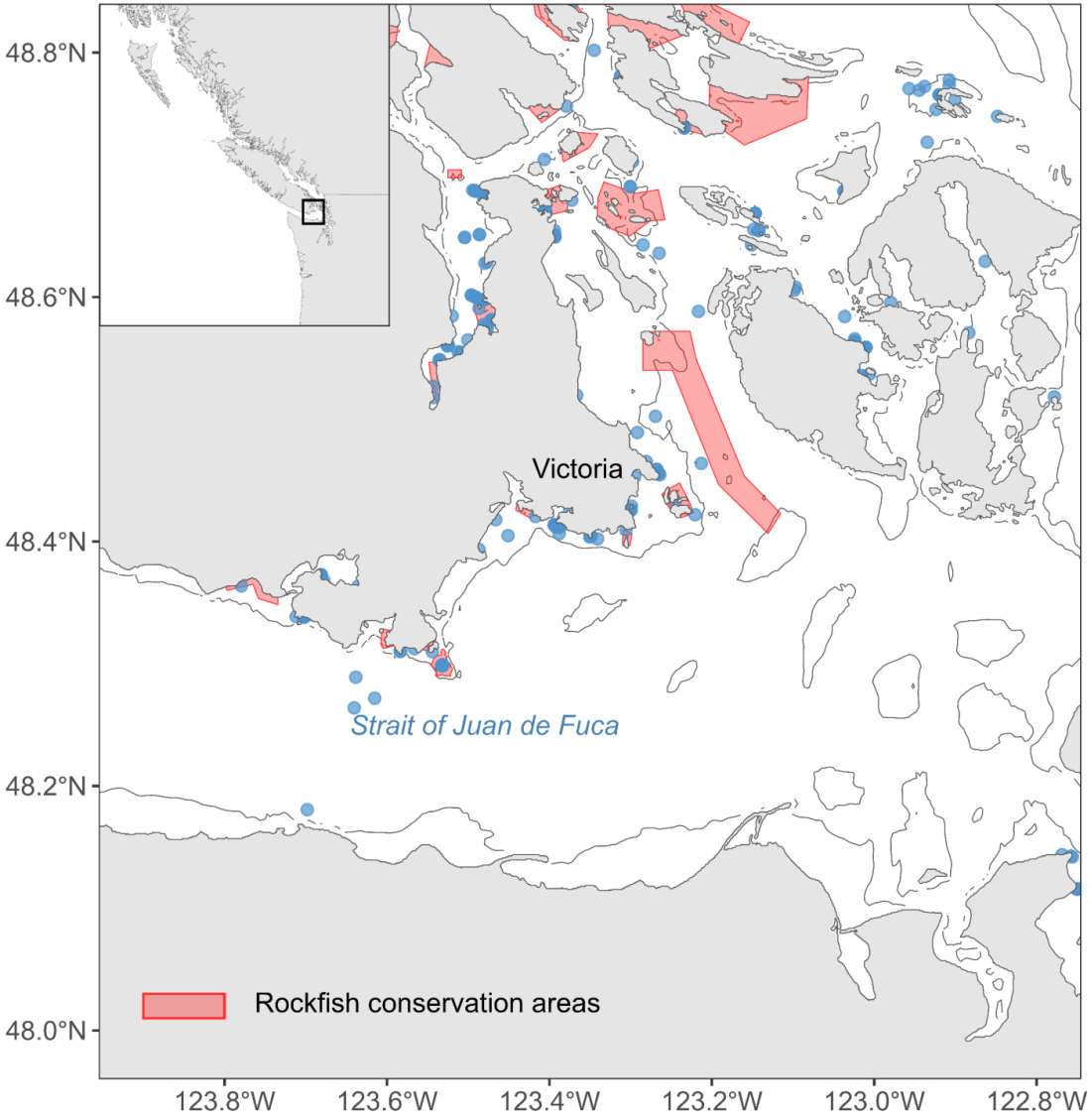


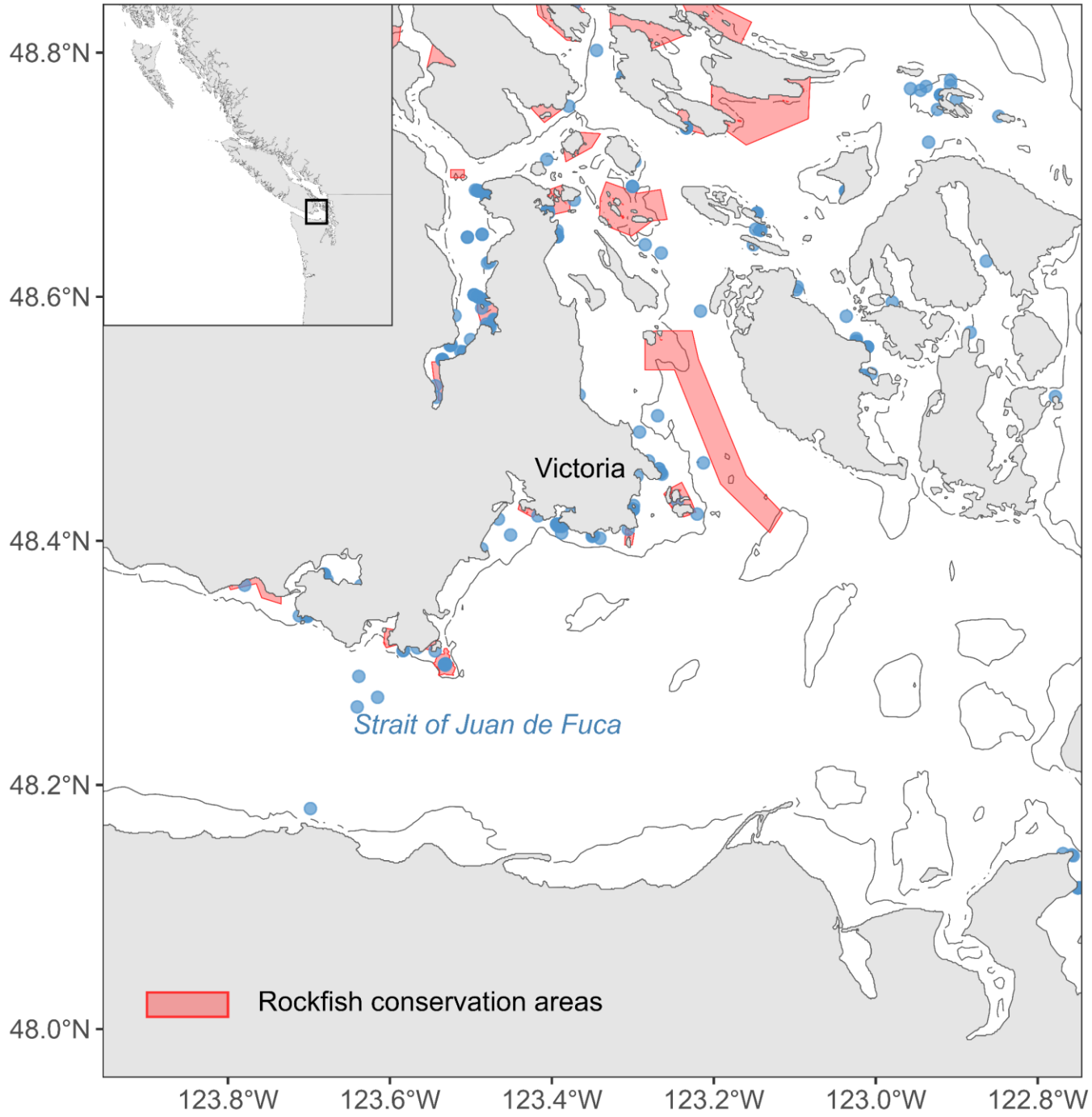
Add inset map, labels



Add 50 metre depth contour

```
geom_spatraster_contour(data = gebco, breaks = c(-50))
```




48.8°N
48.6°N
48.4°N
48.2°N
48.0°N
123.8°W
123.6°W
123.4°W
123.2°W
123.0°W
122.8°W
Victoria
Strait of Juan de Fuca
Rockfish conservation areas

Satellite maps

`basemaps` or **`OpenStreetMap`** packages

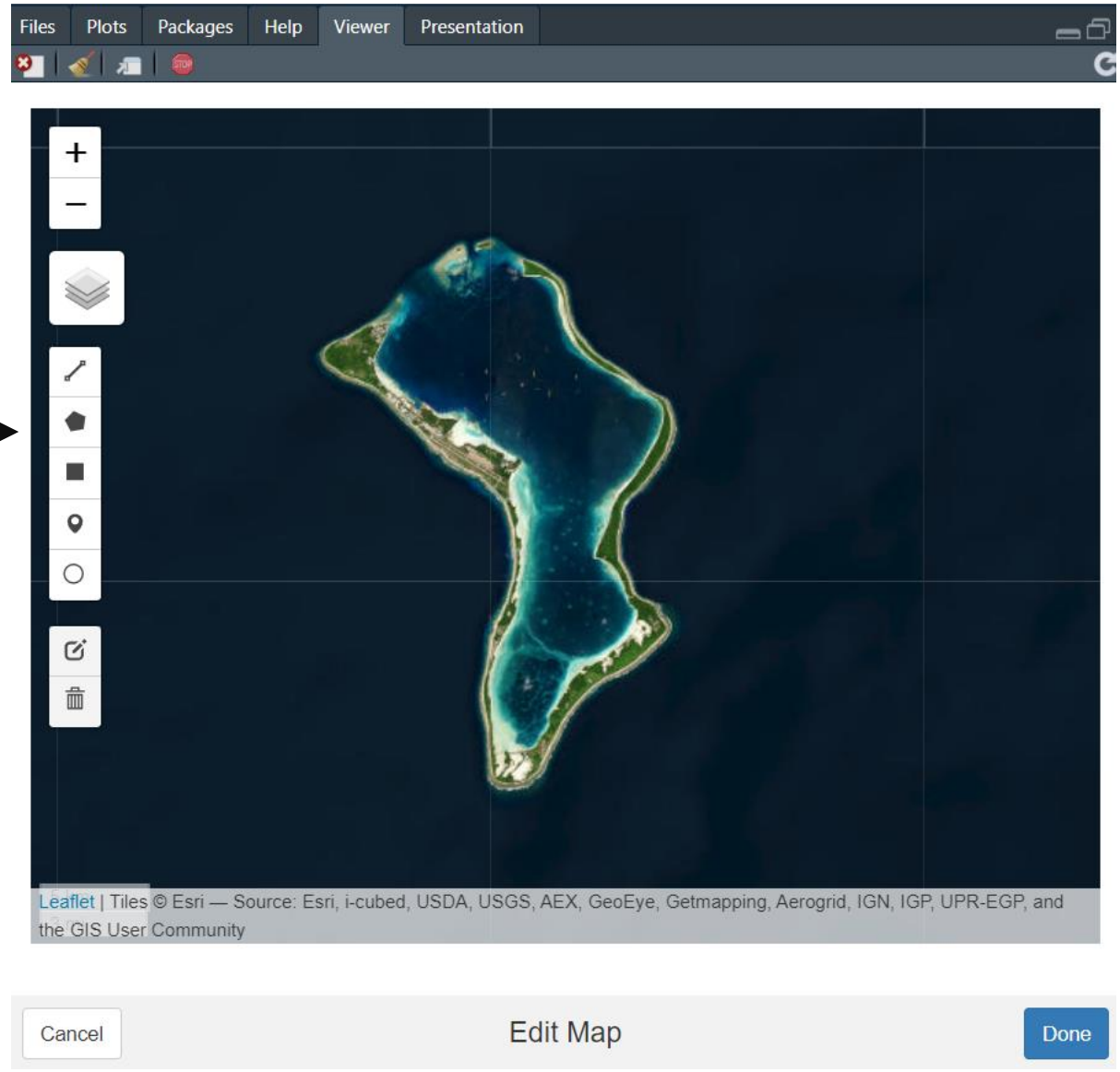
Cool way to show your study area!

Satellite maps

Satellite maps

basemaps package

1. Define bounding box, either with
sf::st_bbox() or **basemaps::draw_ext()**

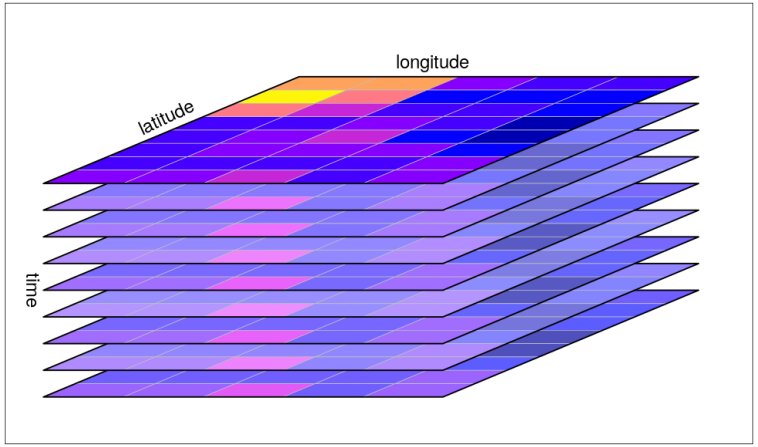



```
bc.box <- st_bbox(c(xmin = -142.3, ymin = 43,
                    xmax = -110.5, ymax = 58),
                  crs = st_crs(4326))
```

Satellite maps

```
bc.box <- st_bbox(c(xmin = -123.95, ymin = 48.1,
                    xmax = -122.9, ymax = 48.7),
                  crs = st_crs(4326))

satellite <- basemap_stars(ext = bc.box,
                           map_service = "esri",
                           type = "world_imagery",
                           map_res = 1)
```

`stars` package: raster cubes



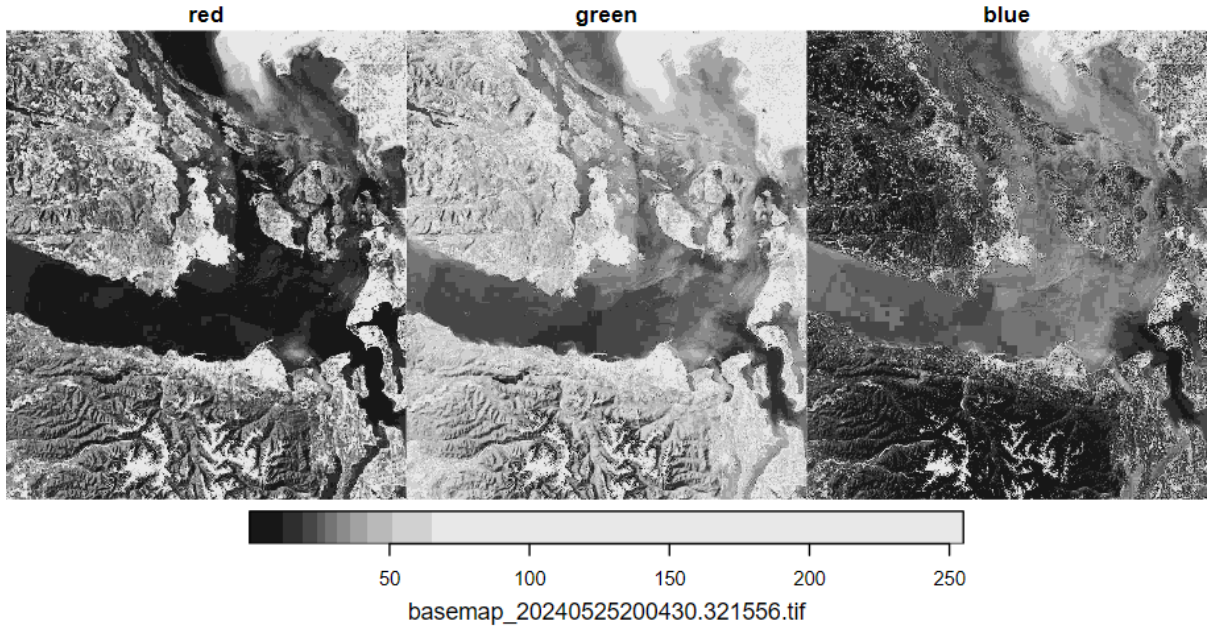
xmin = -123.95, ymin = 48.1, xmax = -122.9, ymax = 48.7

Satellite maps

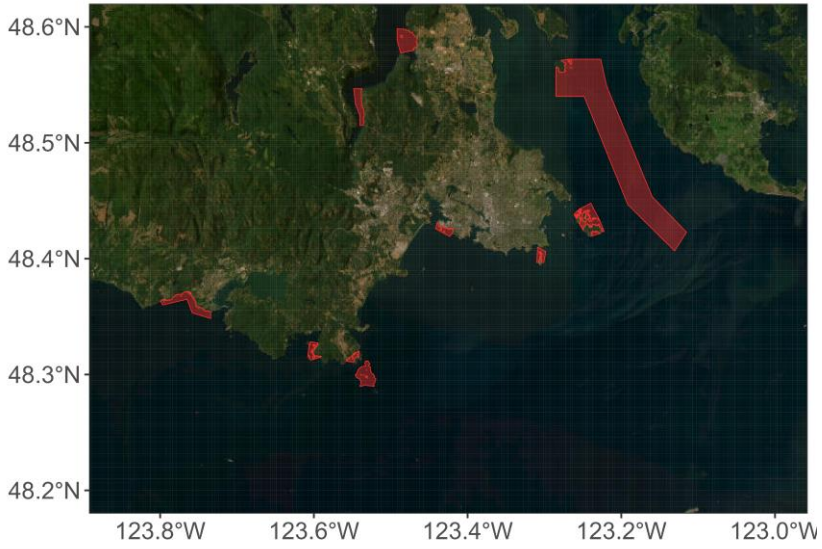
```
satellite <- basemap_stars(ext = bc.box,
                           map_service = "esri",
                           type = "world_imagery",
                           map_res = 1)

satellite.rgb <- st_rgb(satellite)
```

`st_rgb()` converts the satellite imagery into one RGB image

`st_rgb()` converts the satellite imagery into one RGB image



`st_rgb()`



Satellite maps

```
satellite <- basemap_stars(ext = bc.box,
                           map_service = "esri",
                           type = "world_imagery",
                           map_res = 1)

st_crs(satellite)
```

Satellite basemap is in Web Mercator (EPSG 3857)

Can transform to WGS 84 but reduces quality.

Satellite maps

```
satellite <- basemap_stars(ext = bc.box,
                           map_service = "esri",
                           type = "world_imagery",
                           map_res = 1)

satellite.rgb <- st_rgb(satellite)
```

The basemap was initially in Web Mercator (EPSG 3857), so we convert it to WGS 84 (EPSG 4326)

Same CRS: compatible with other layers

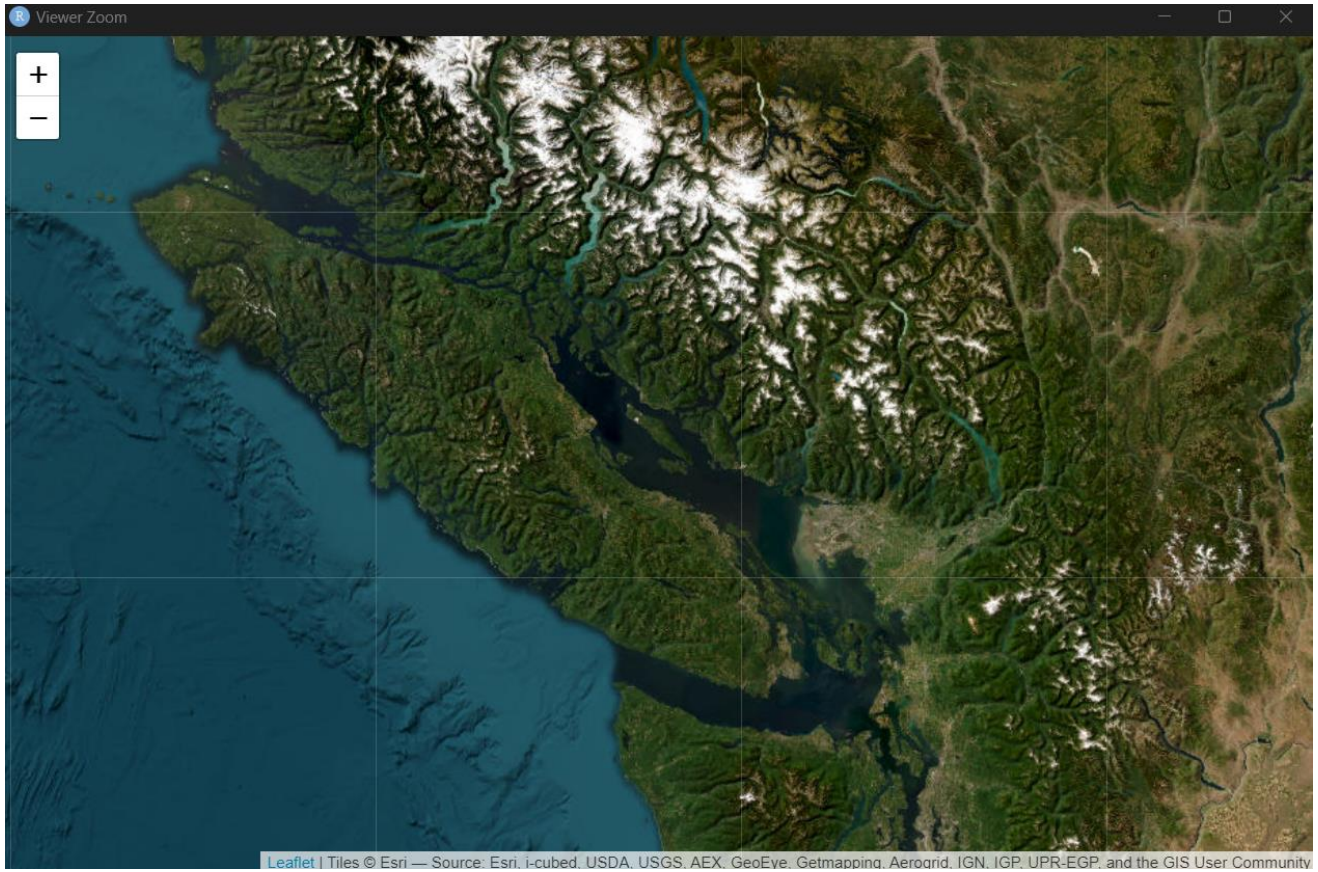
Satellite maps

```
ggplot() +
       geom_stars(data = satellite.rgb) +
       geom_sf(data = rca.wm)
```

Interactive maps with **leaflet**

leaflet is a javascript library to make interactive maps

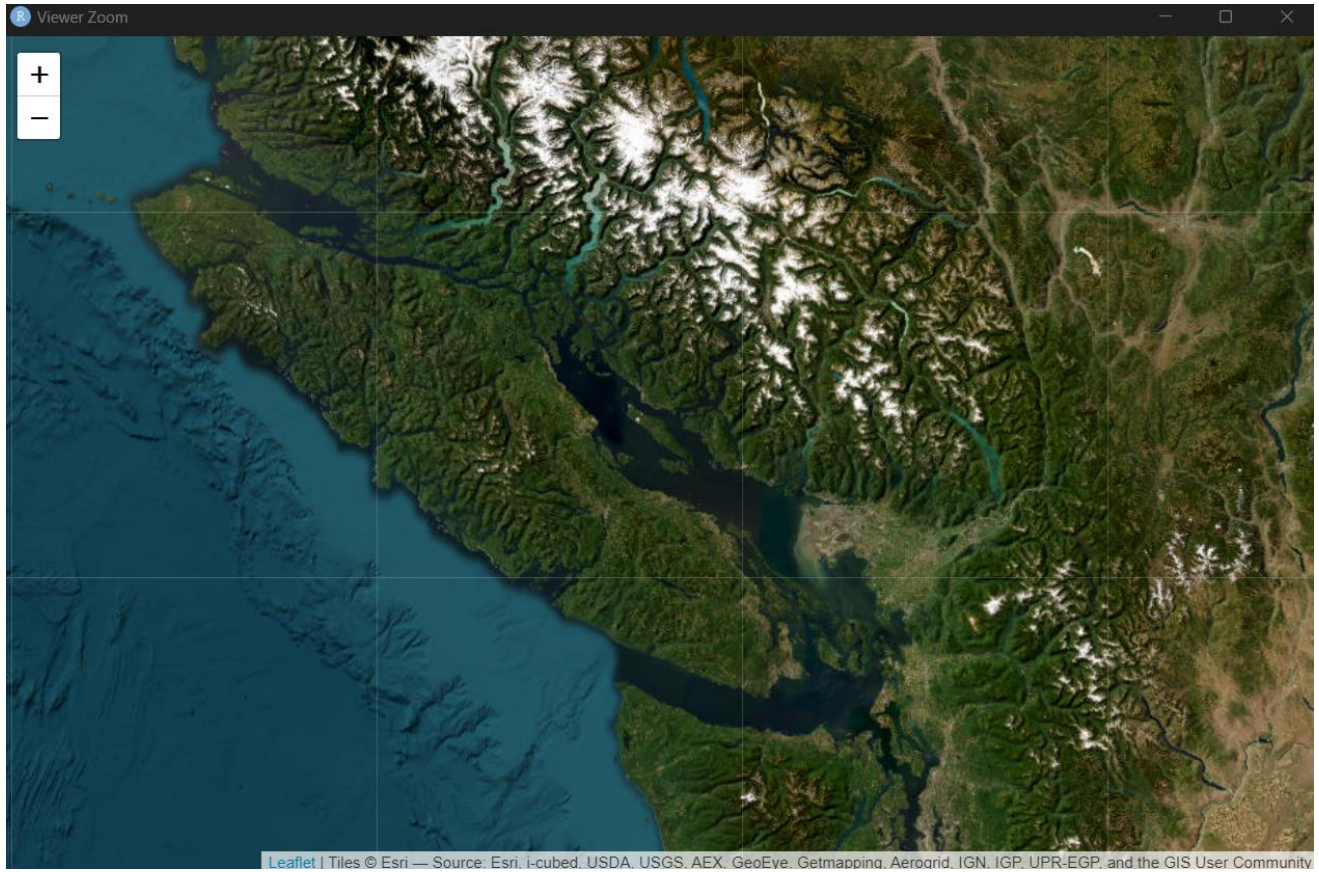
The **leaflet** package allows us to make these maps in R



Interactive maps with **leaflet**

leaflet is a javascript library to make interactive maps

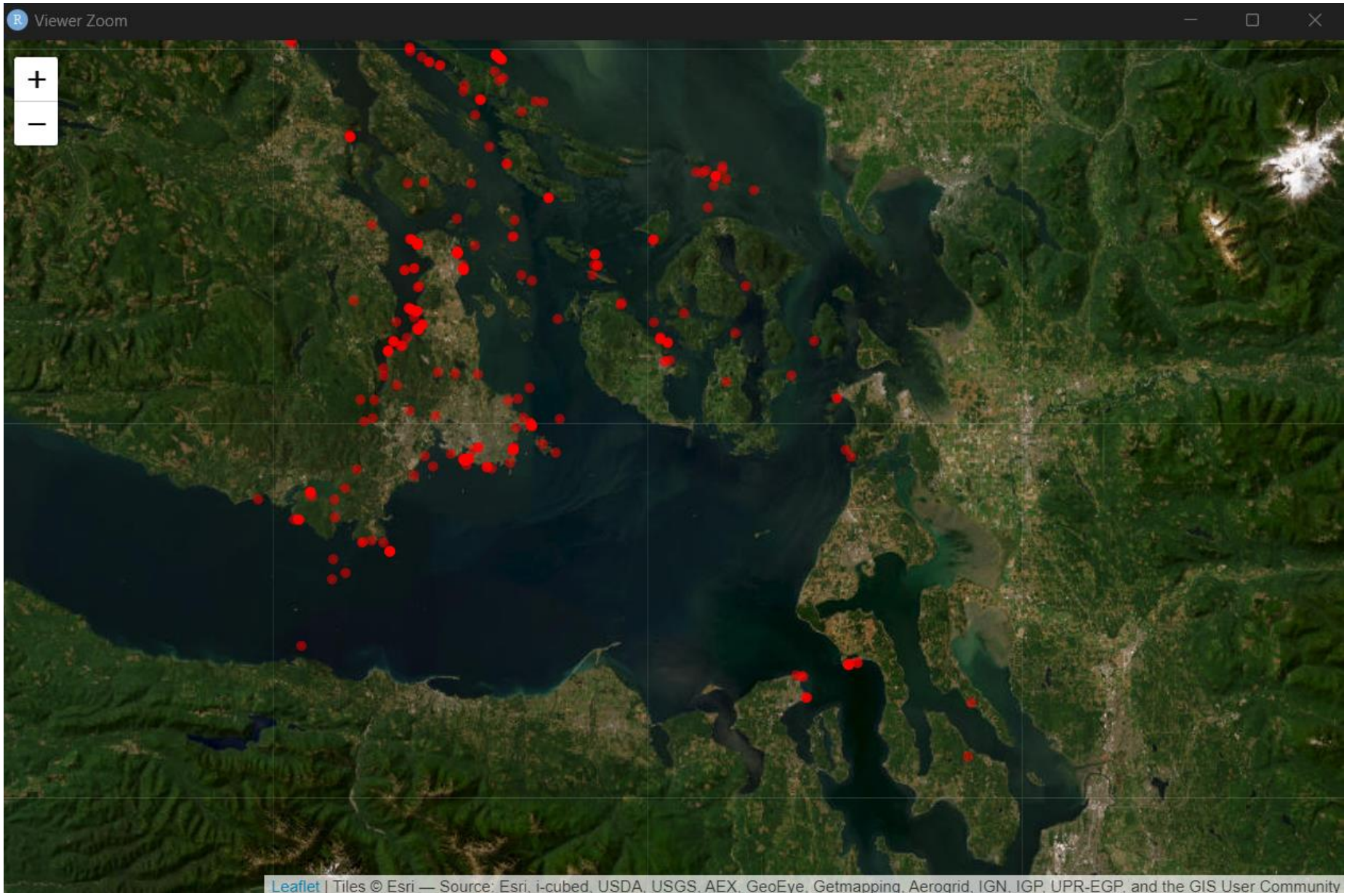
The **leaflet** package allows us to make these maps in R



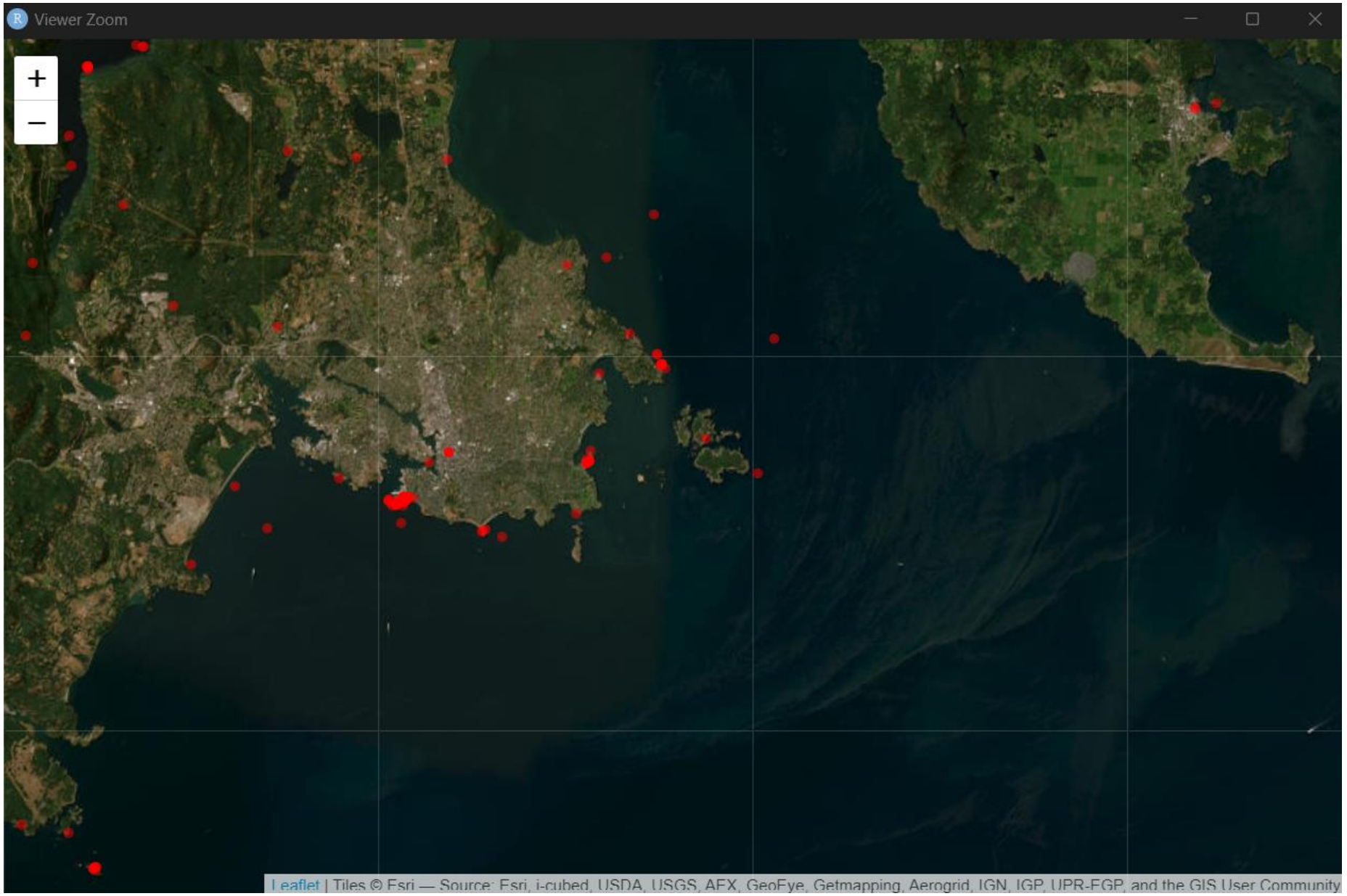
https://rstudio.github.io/leaflet/

Note: current incompatibility with R 4.4:
https://github.com/rstudio/rstudio/issues/14603

Interactive maps with **leaflet**

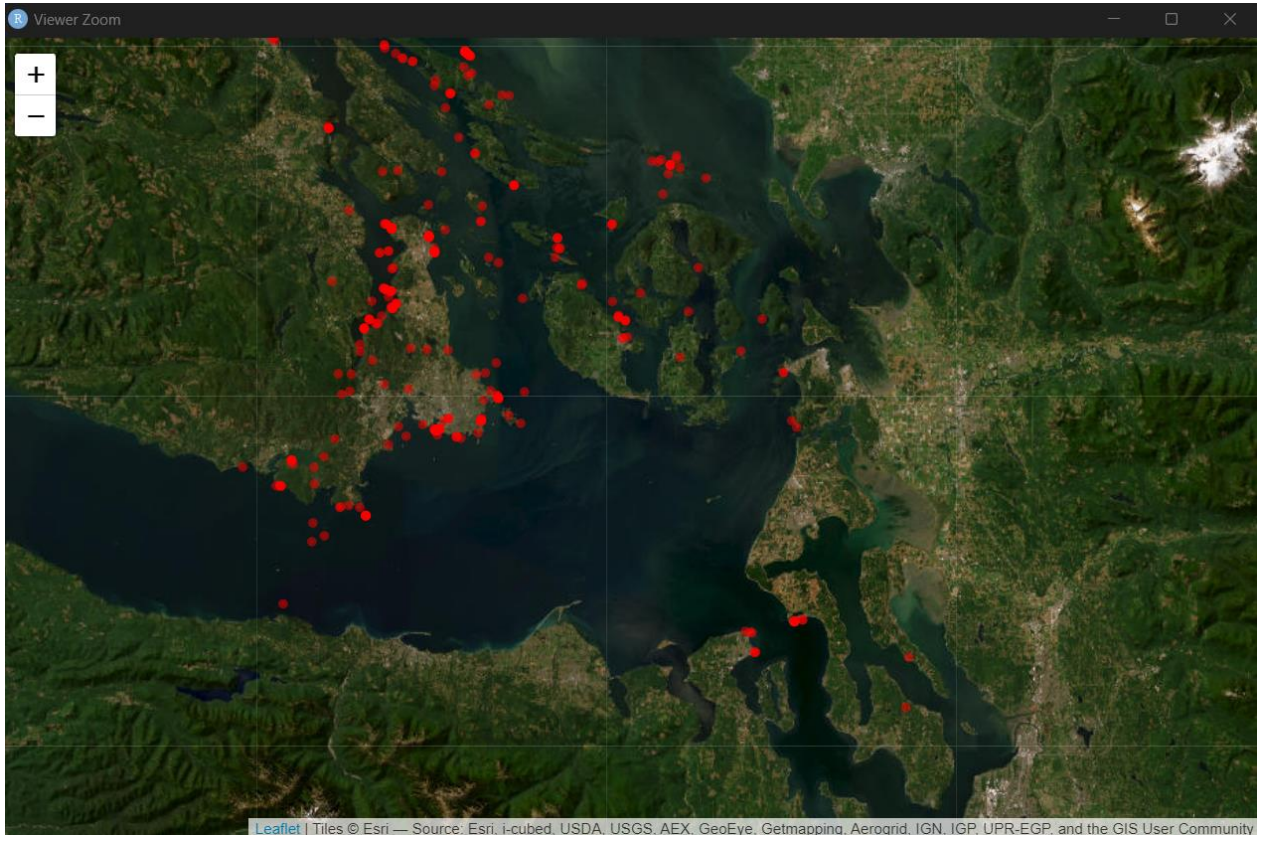


Interactive maps with **leaflet**



Interactive maps with **leaflet**

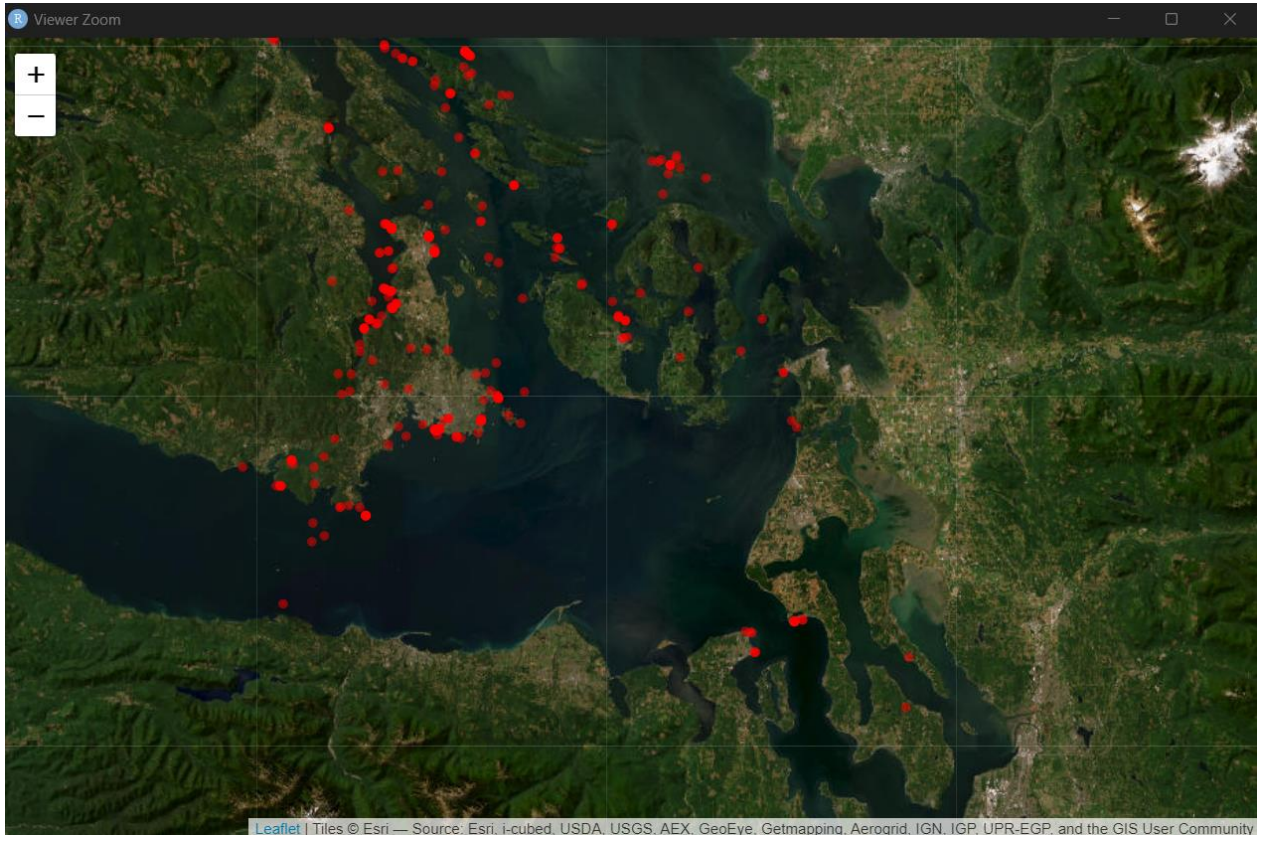
```
leaflet() |>
      setView(lng = -123, lat = 48.4, zoom = 9) |>
      addCircles(data = rockfish,
                 lng = rockfish$longitude,
                 lat = rockfish$latitude) |>
      addProviderTiles(providers$Esri.WorldImagery)
```



Interactive maps with **leaflet**

```
leaflet() |>
      setView(lng = -123, lat = 48.4, zoom = 9) |>
      addCircles(data = rockfish,
                 lng = rockfish$longitude,
                 lat = rockfish$latitude) |>
      addProviderTiles(providers$Esri.WorldImagery)
```

|> strings together multiple functions



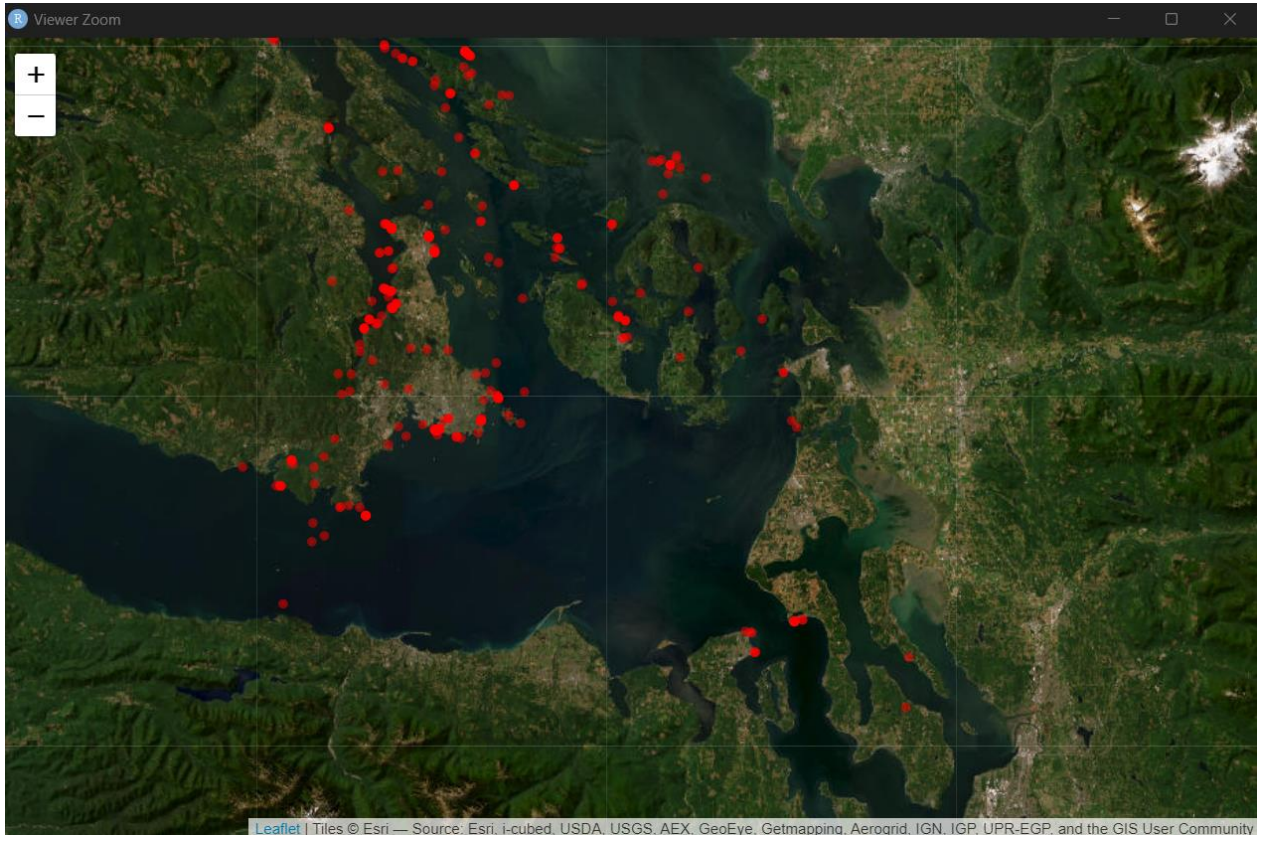
Interactive maps with **leaflet**

```
leaflet() |>
      setView(lng = -123, lat = 48.4, zoom = 9) |>
      addCircles(data = rockfish,
                 lng = rockfish$longitude,
                 lat = rockfish$latitude) |>
      addProviderTiles(providers$Esri.WorldImagery)
```

|> strings together multiple functions

baseR version of >%>

“pipe”



Animated maps

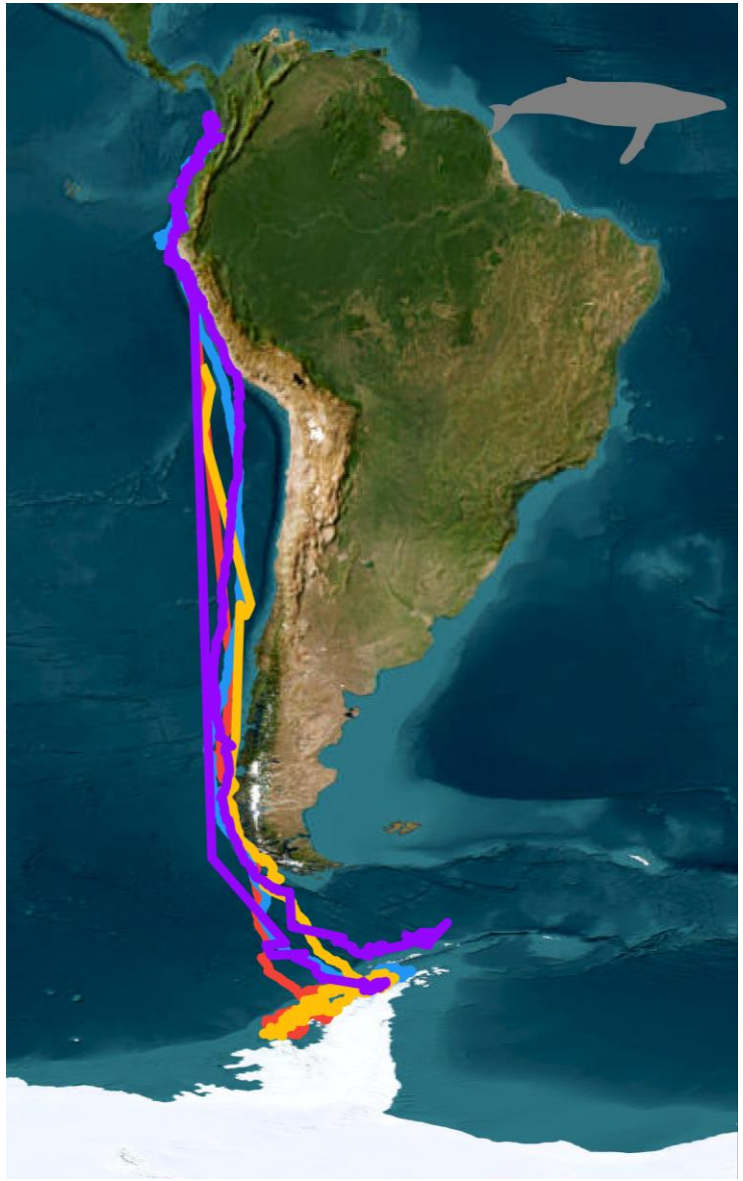
`gganimate` package

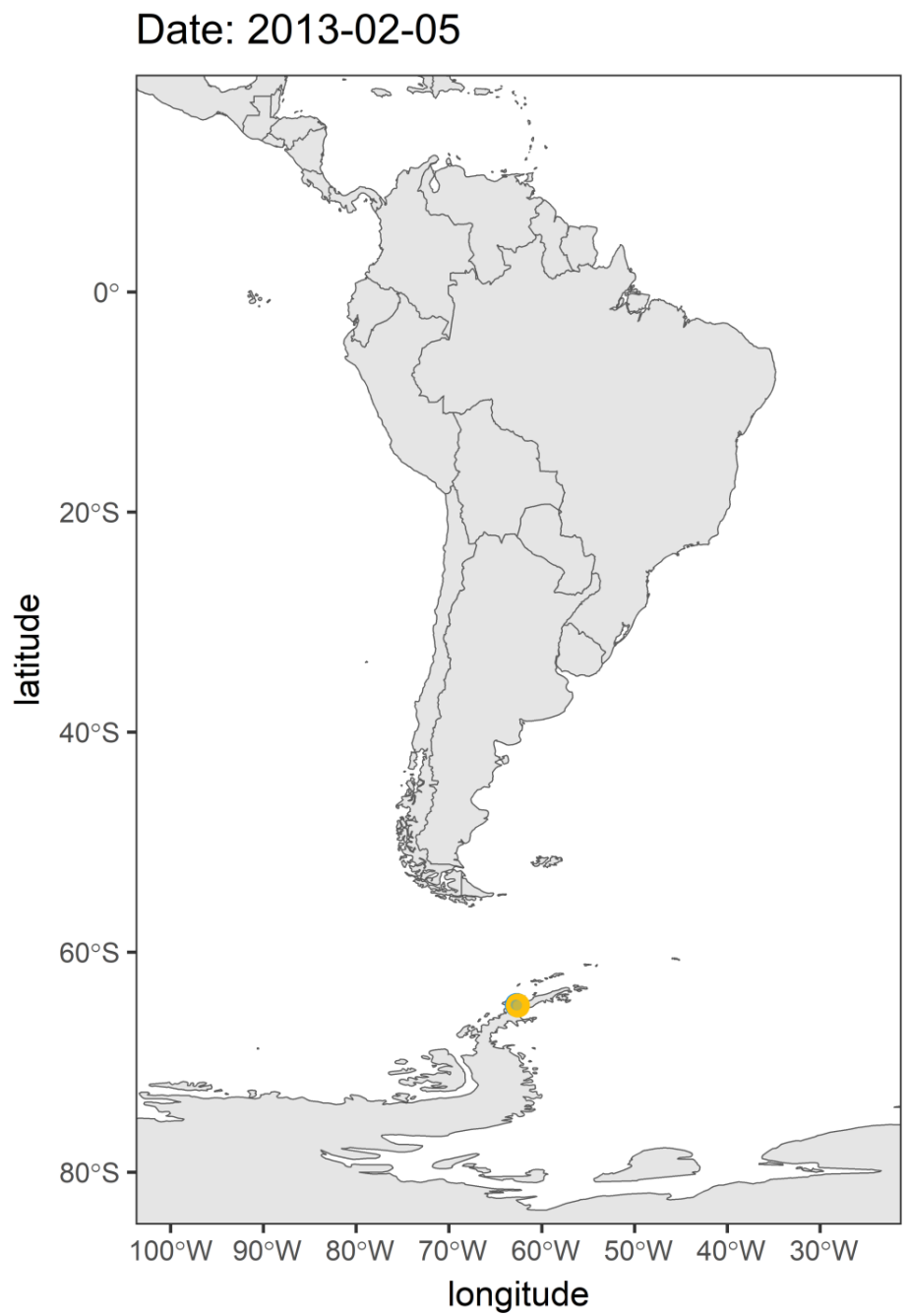
Exciting tool to visualize how a system changes over time, or track movements of tagged animals

Animated maps

Date: 2023-01-15

Humpback whale migrations





Humpback whale data

Satellite tags placed on humpback whales in 2013

RESEARCH Open Access

Capturing foraging and resting behavior using nested multivariate Markov models in an air-breathing marine vertebrate

Ben G. Weinstein[1*], Ladd Irvine[1] and Ari S. Friedlaender[1,2]

RESEARCH Open Access

First description of migratory behavior of humpback whales from an Antarctic feeding ground to a tropical calving ground

Michelle Modest[1*], Ladd Irvine[2], Virginia Andrews-Goff[3], William Gough[4], David Johnston[5], Douglas Nowacek[5], Logan Pallin[1], Andrew Read[5], Reny Tyson Moore[6] and Ari Friedlaender[1]

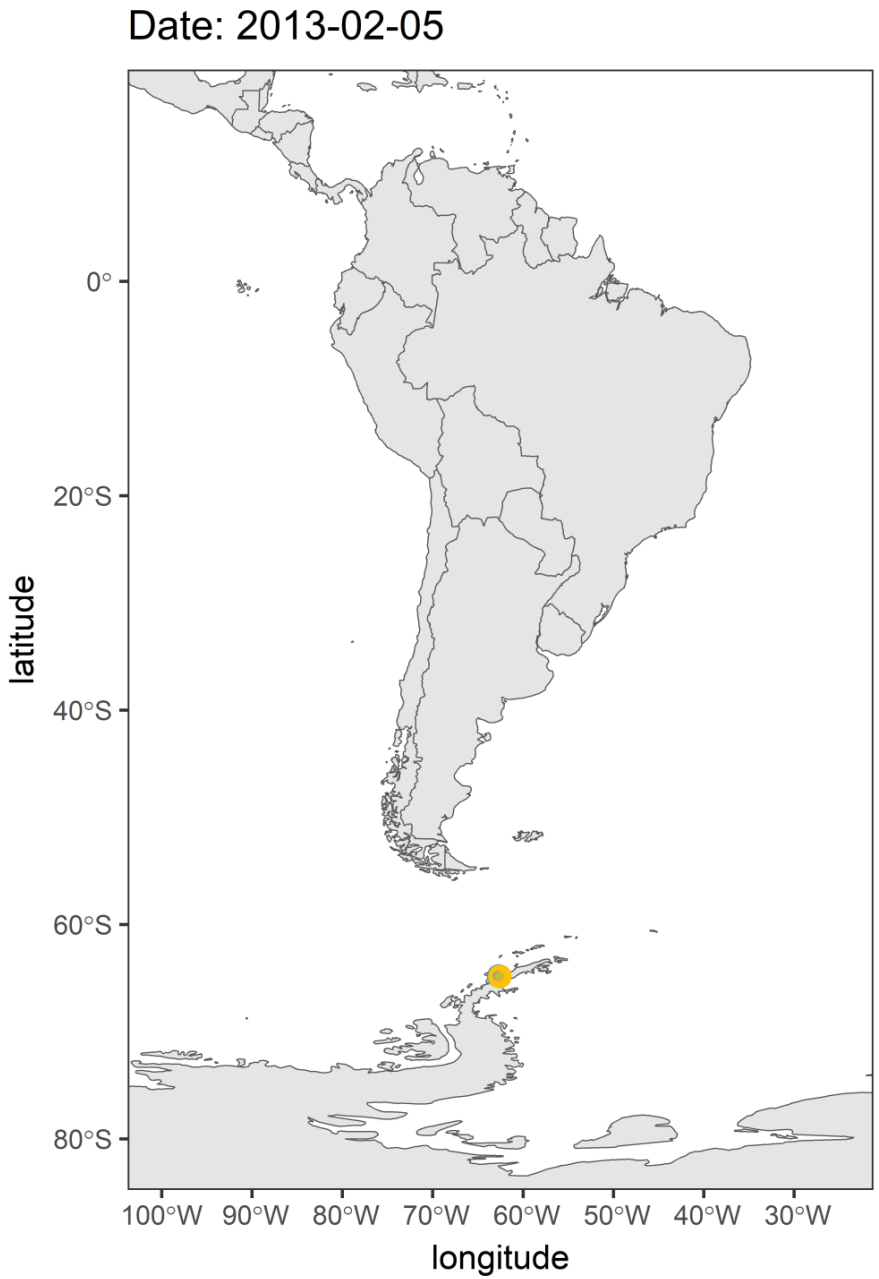

NOAA Fisheries


National Geographic

Animated maps

1. Make static map with ggplot()

2. Add animation with gganimate functions



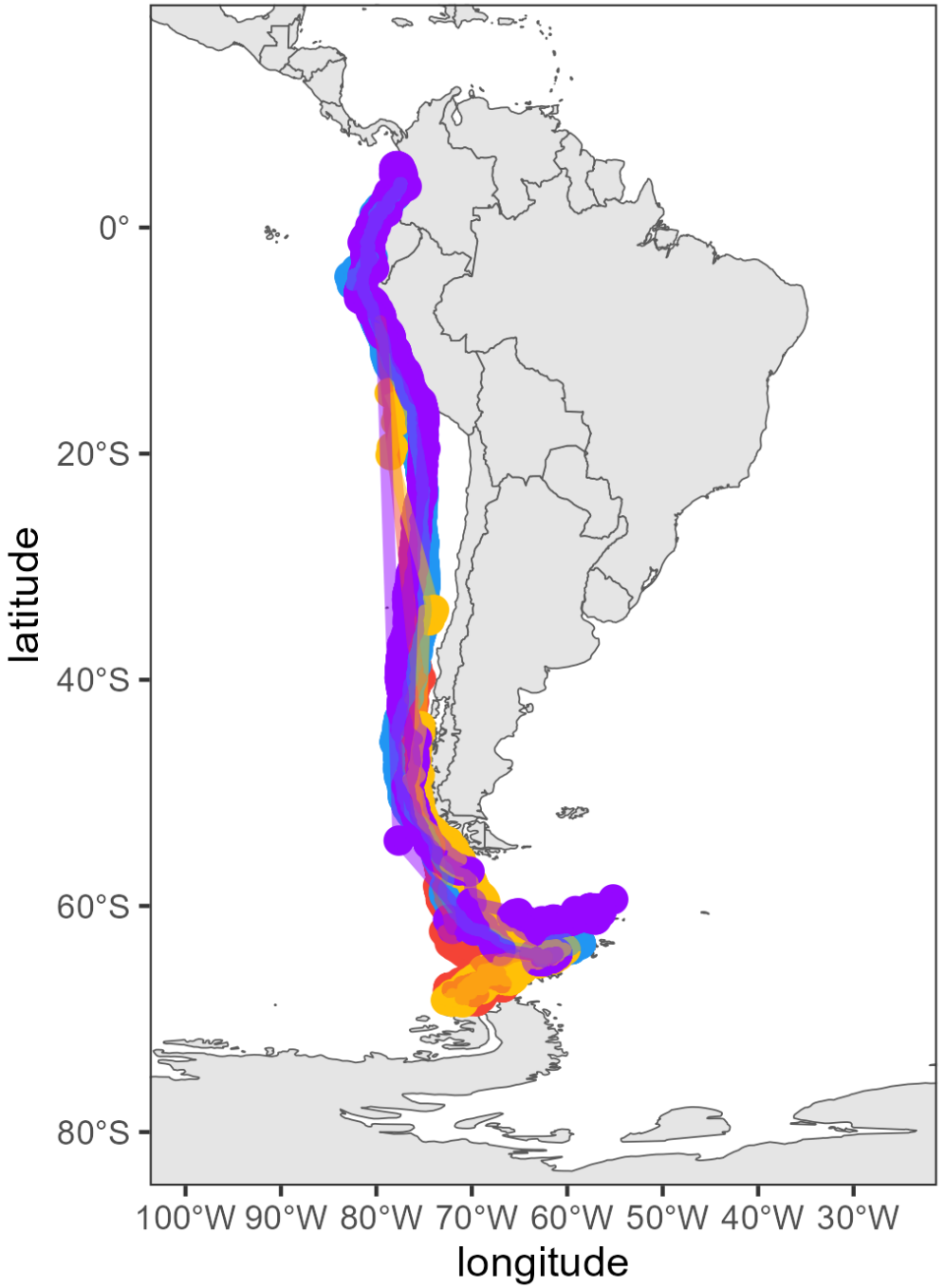
Animated maps

1. Static map

```
static <- ggplot() +
  geom_sf(data = south.america.antarctica) +
  geom_point(data = humpbacks,
             aes(x = longitude, y = latitude,
             group = animal_id) +
  geom_point(data = humpbacks,
             aes(x = longitude, y = latitude,
             group = animal_id)
```

`geom_point()`: all satellite tag locations

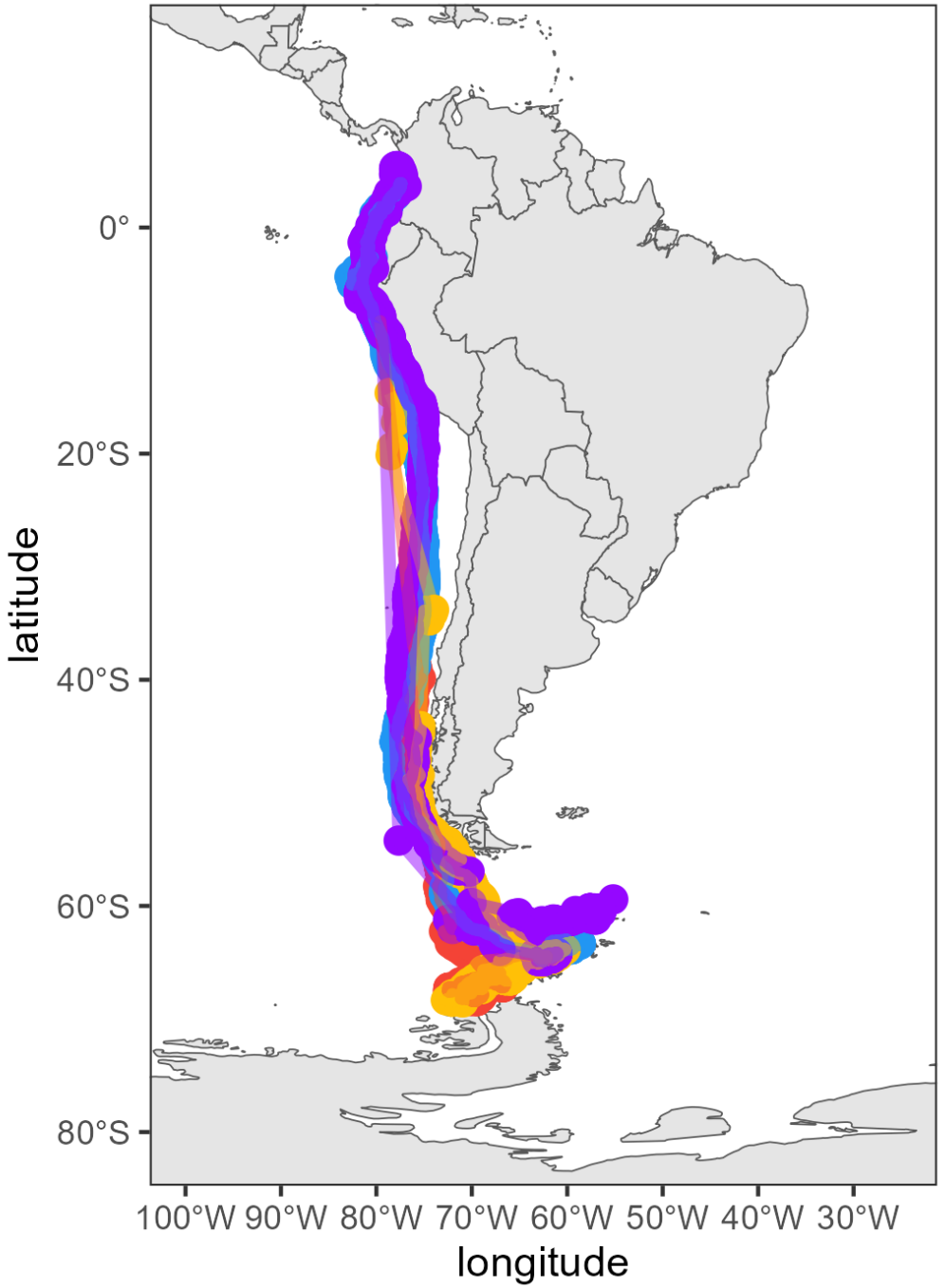
`geom_path()`: connects successive locations



Animated maps

1. Static map

```
static <- ggplot() +
  geom_sf(data = south.america.antarctica) +
  geom_point(data = humpbacks,
             aes(x = longitude, y = latitude,
             group = animal_id) +
  geom_point(data = humpbacks,
             aes(x = longitude, y = latitude,
             group = animal_id)
```



`gganimate` recognizes `geom_point()` and `geom_path()` together

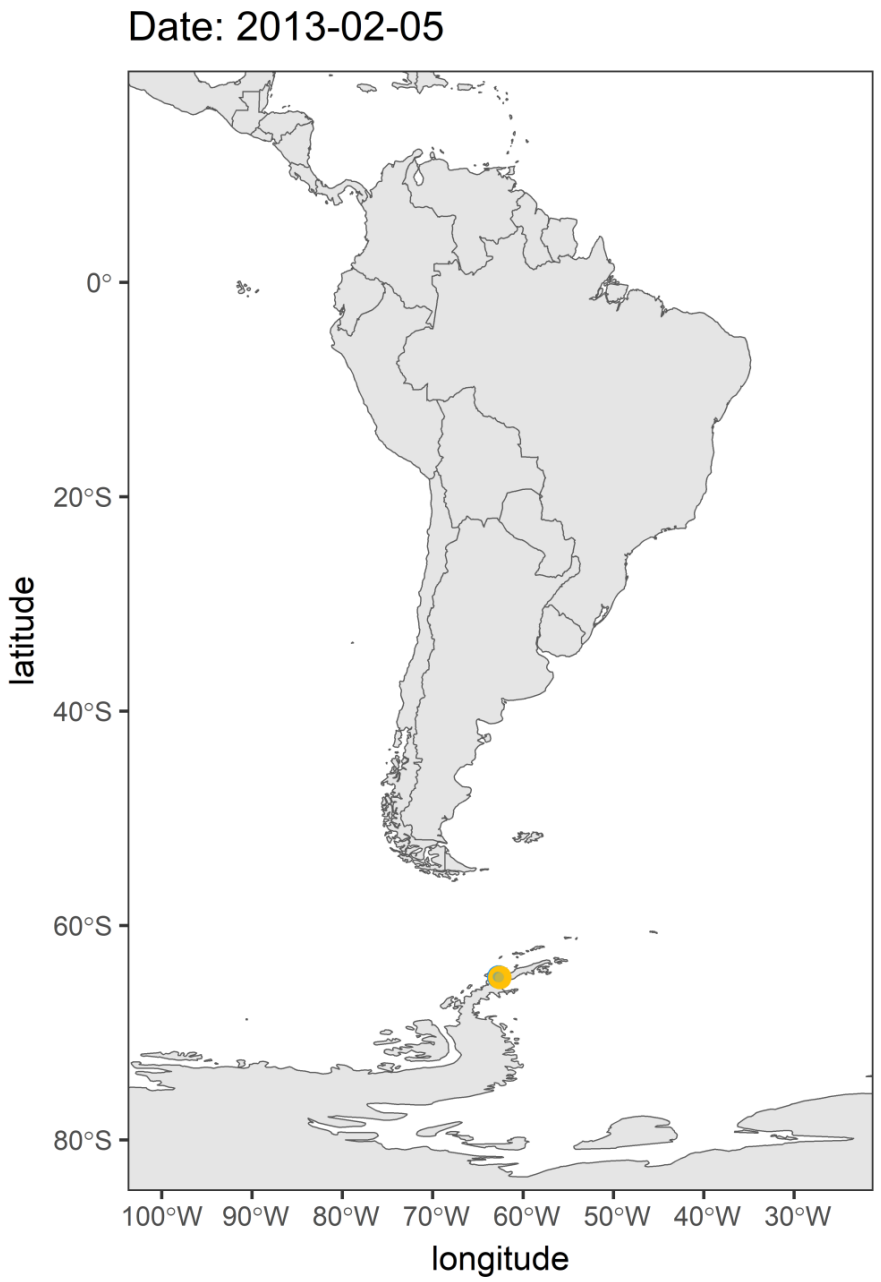
Plots the entire track path to date, but only the current point

Animated maps

1. Make static map with ggplot()

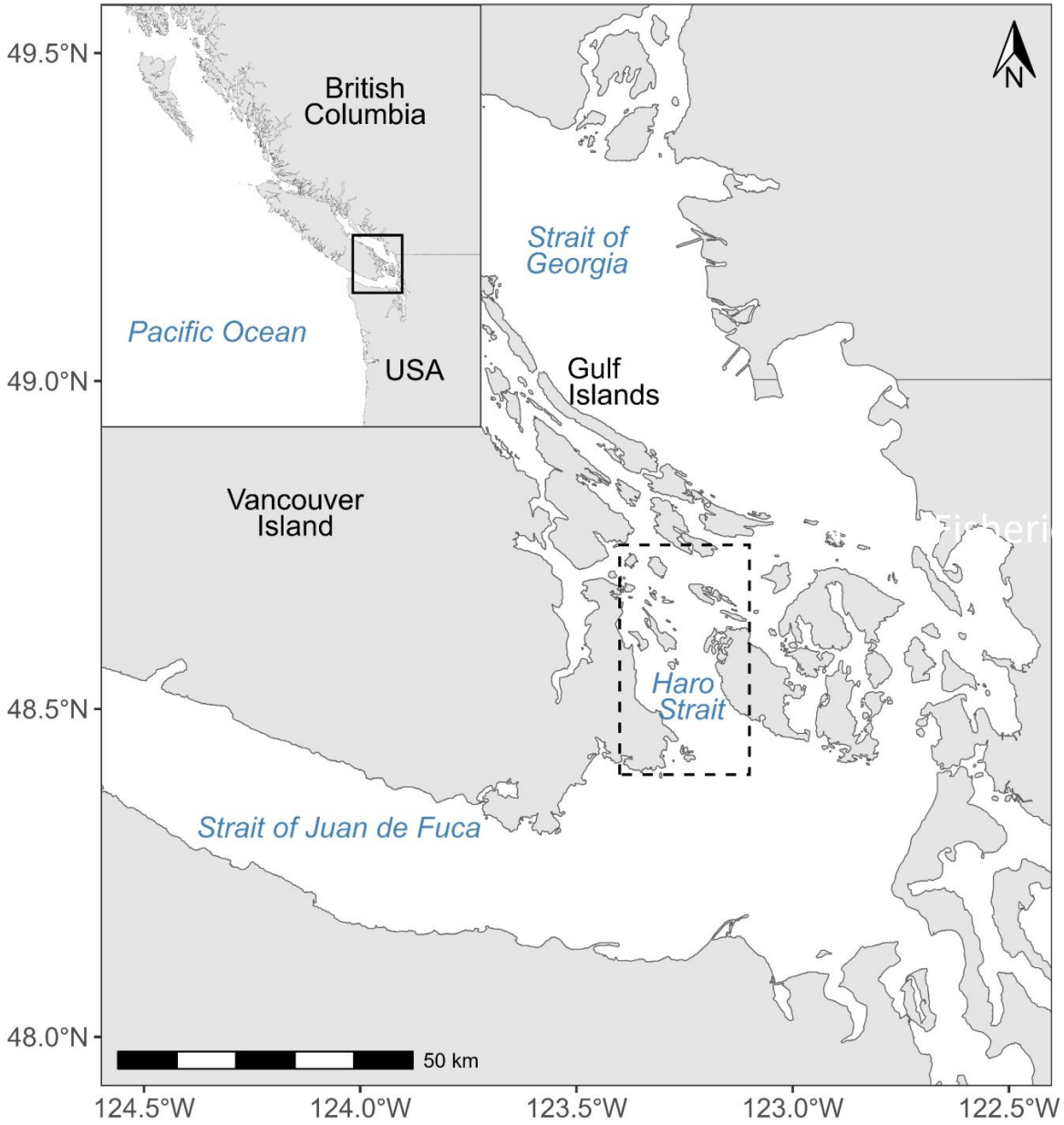
2. Add animation with gganimate functions

```
dynamic <- static +
   transition_reveal(datetime) +
   labs(title = "Date:as_date(frame_along)}") +
   ease_aes("linear")

anim_save("figures/whale-animation.gif", dynamic
```



Key Points

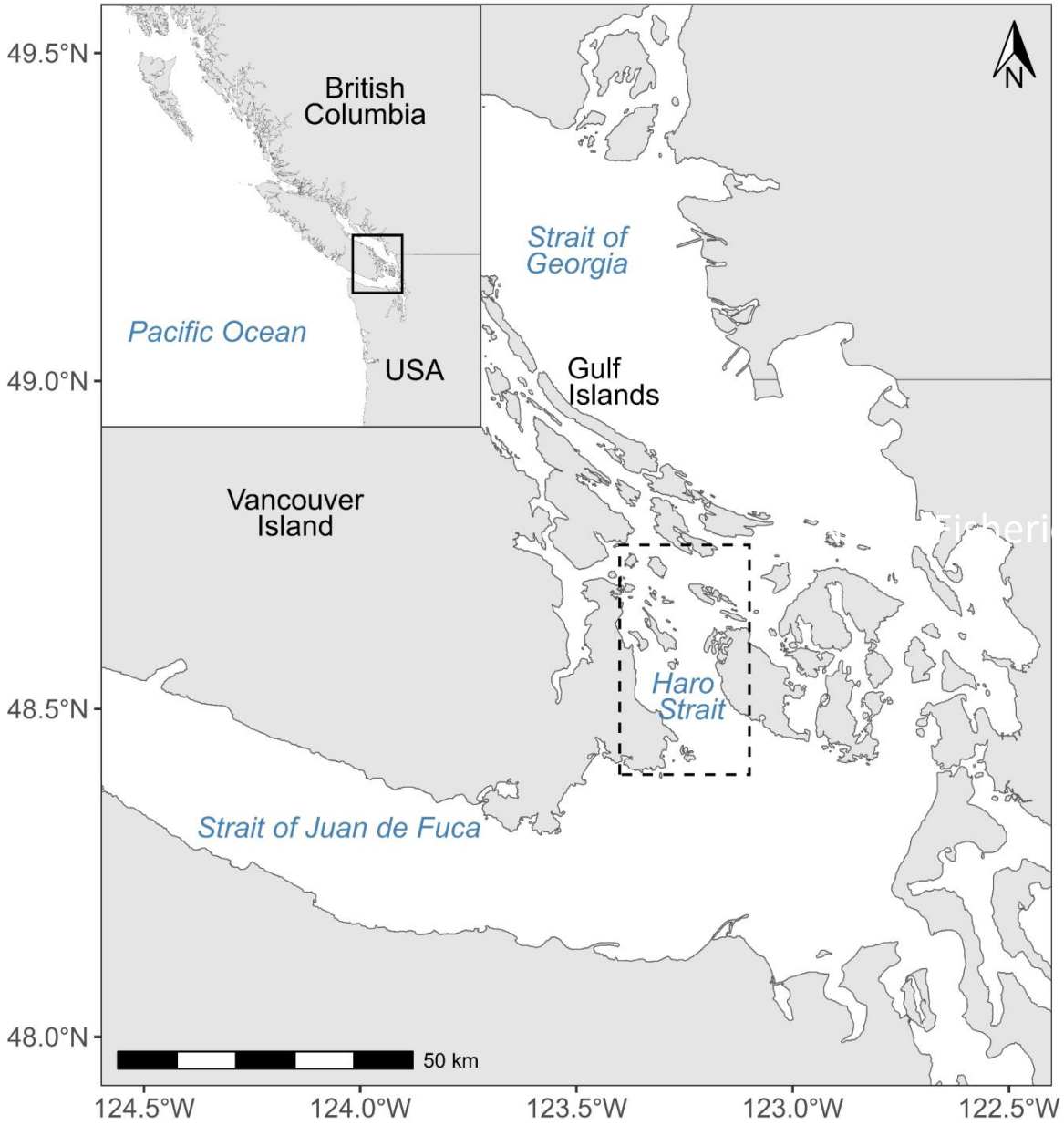
1. **Study area maps:**
    a. high resolution
    b. show what's important
    c. clean

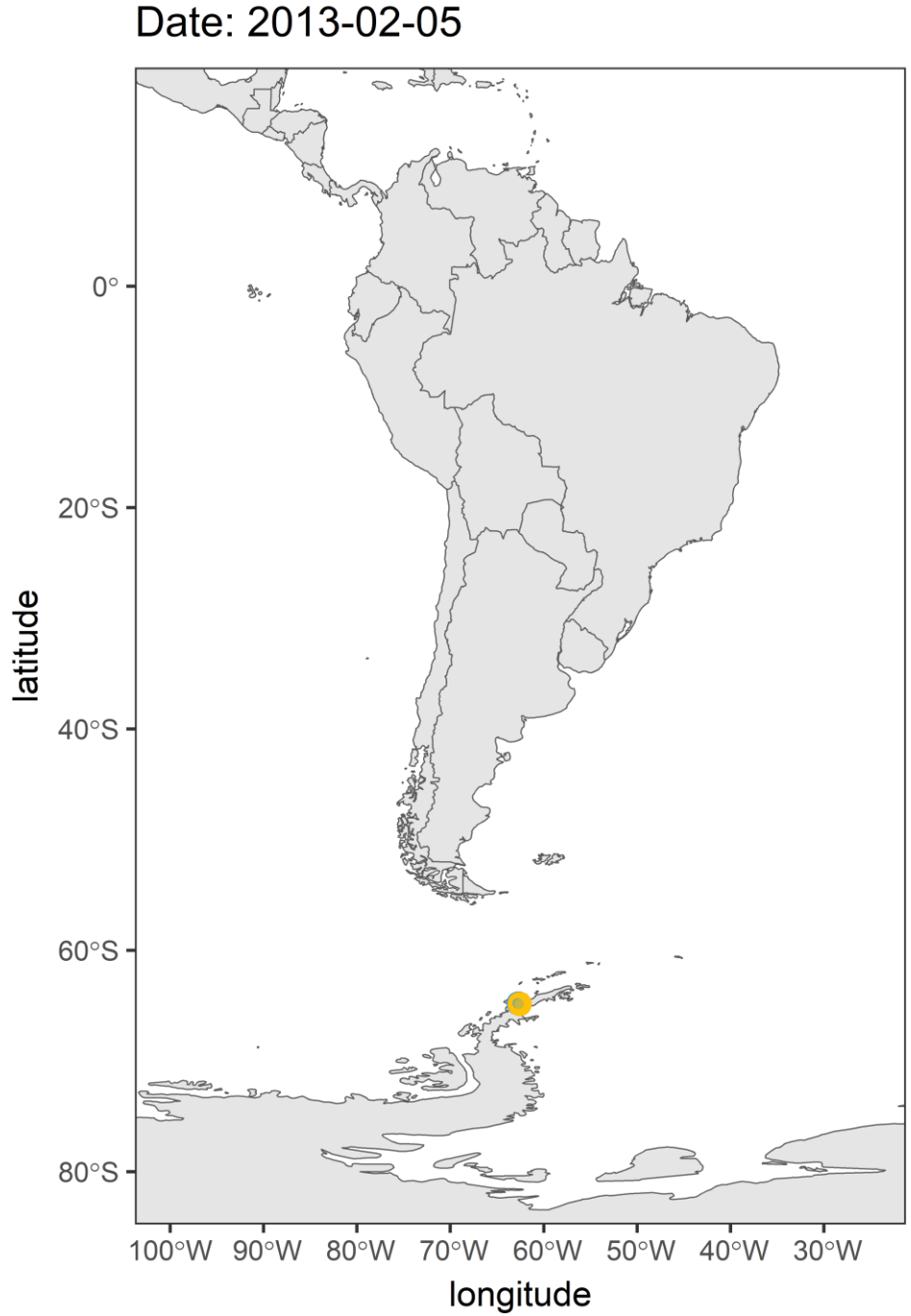


Key Points

1. **Study area maps:**
    a. high resolution
    b. show what's important
    c. clean

2. R can make lots of different maps
    a. bathymetry, topography
    b. satellite maps
    c. interactive maps
    d. animated maps




Date: 2013-02-05
0°
20°S
40°S
60°S
80°S
latitude
100°W
90°W
80°W
70°W
60°W
50°W
40°W
30°W
longitude

Acknowledgements

CSEE Organizing Committee

Canadian Institute of Ecology and Evolution

Francis Juanes, Will Duguid, and Juanes/Baum labmates

University of Victoria

Juanes Lab

PACIFIC SALMON FOUNDATION

cjee icée

canadian institute of ecology and evolution
institut canadien d'écologie et d'évolution

Thank you!

Questions/comments: **wgreentree@outlook.com**

Feel free to reach out with any R spatial questions you have in the future! Happy to give R spatial workshops to other groups!

Enjoy the rest of CSEE!

Let’s make our own maps!