

Mapping and analyzing spatial data with R

Wesley Greentree
Pacific Ecology and Evolution Conference
April 1, 2023



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



ArcGIS
QGIS



Territory acknowledgements

- **SFU:** Musqueam, Squamish, Tsleil-Waututh, Katzie, Kwikwetlem, Qayqayt, Kwantlen, Semiahmoo, and Tsawwassen peoples
- **UVic:** We acknowledge and respect the *lək'ʷən̓ən* peoples on whose traditional territory the university stands and the Songhees, Esquimalt and *WSÁNEĆ* peoples whose historical relationships with the land continue to this day.
- **MSc field work:** K'ómoks First Nation



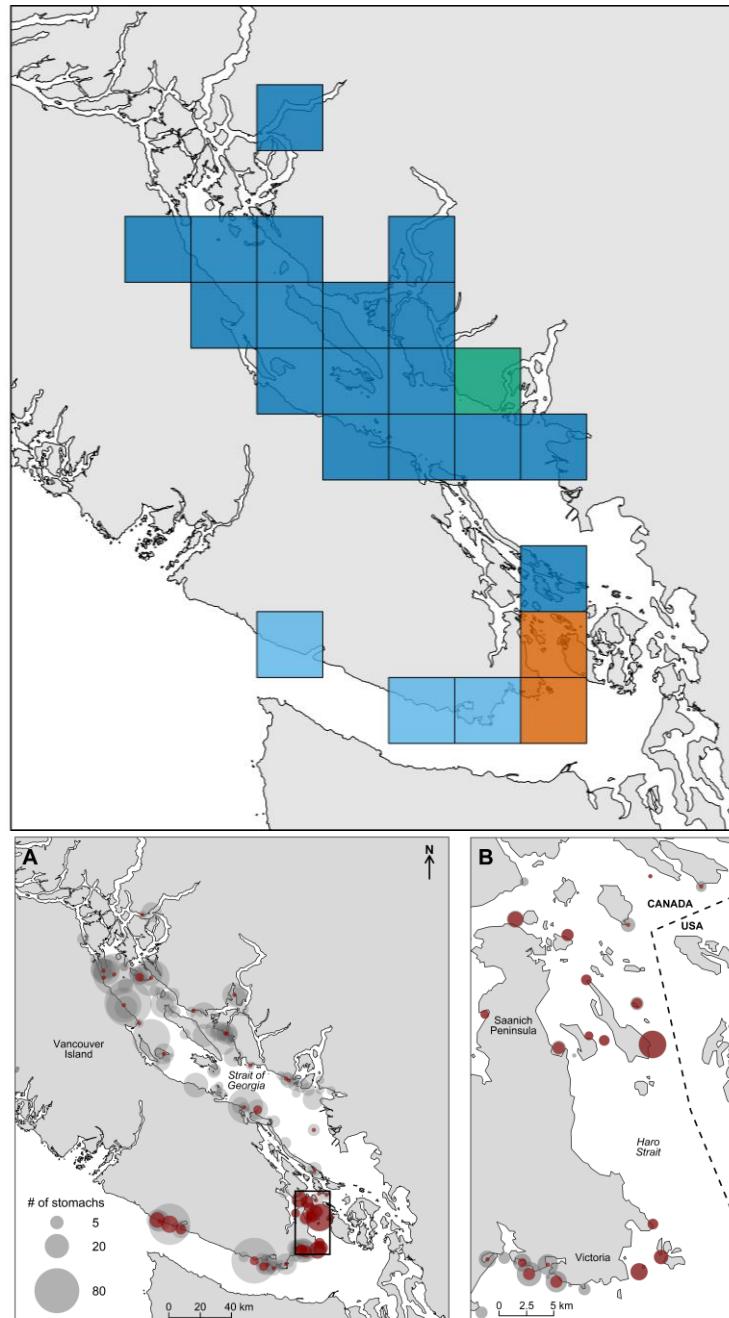


Wesley Greentree (he/him)
MSc student, UVic



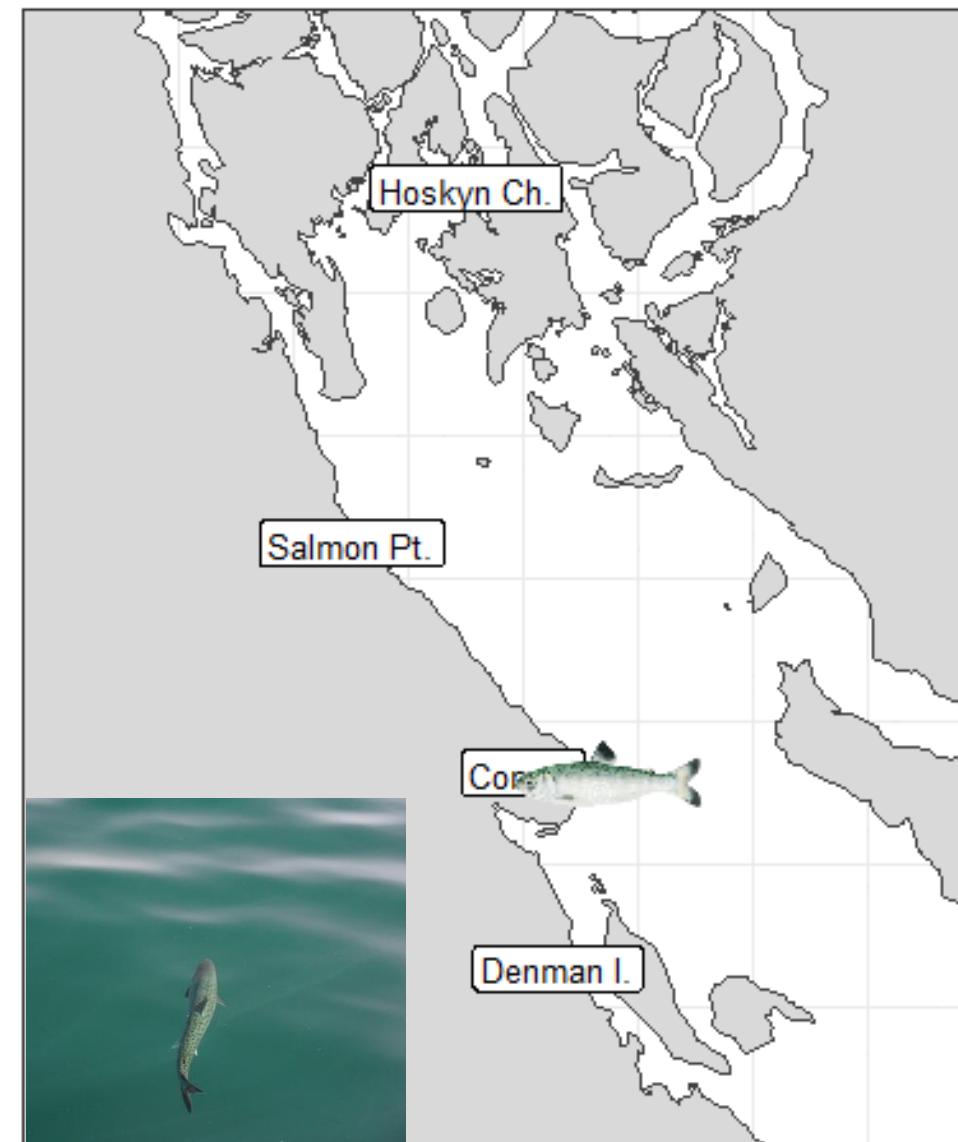


Wesley Greentree (he/him)
MSc student, UVic



Regional food webs

DateTime: 2023-01-22 11:48:03



Salmon migrations

Workshop plan

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

Topics:

- 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:

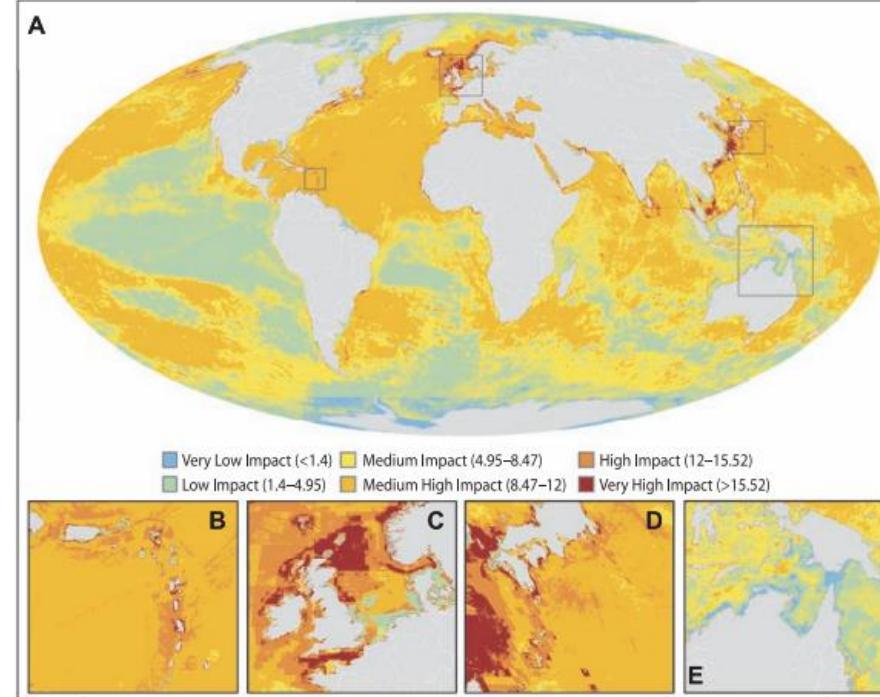
<https://github.com/wesleygreentree/PEEC2023-R-workshop>

Zoom link for viewing live code (copy paste from Github README):

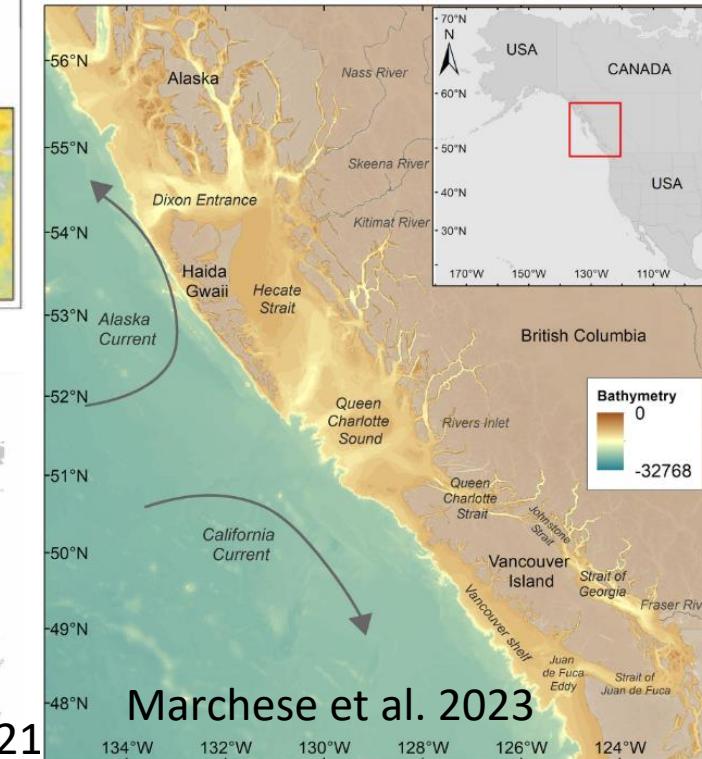
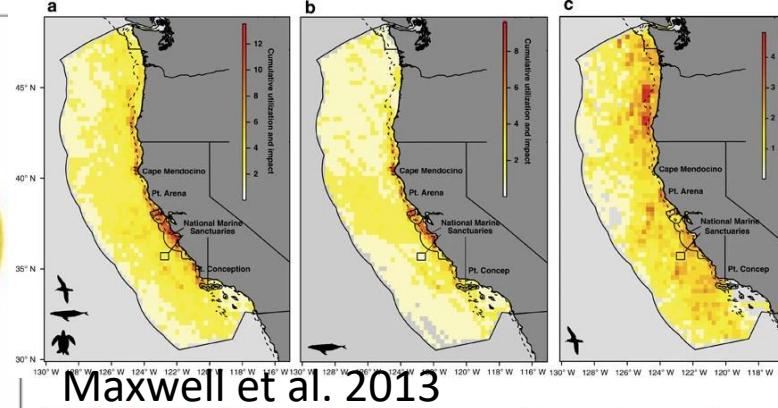
<https://uvic.zoom.us/j/81506791814?pwd=cUkrb1ZlKzFjOW52bVFuS3YzS3JKUT09>

Spatial data are common in ecology

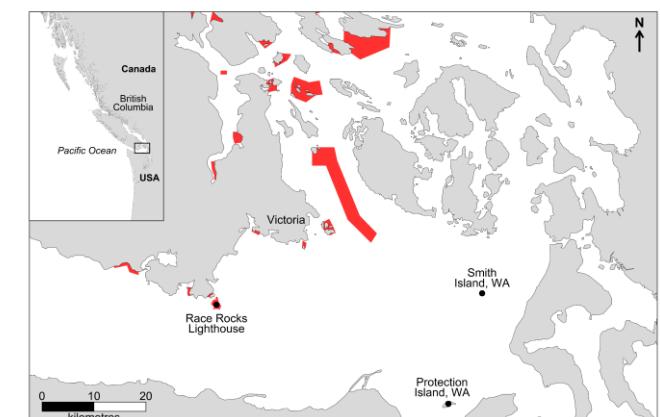
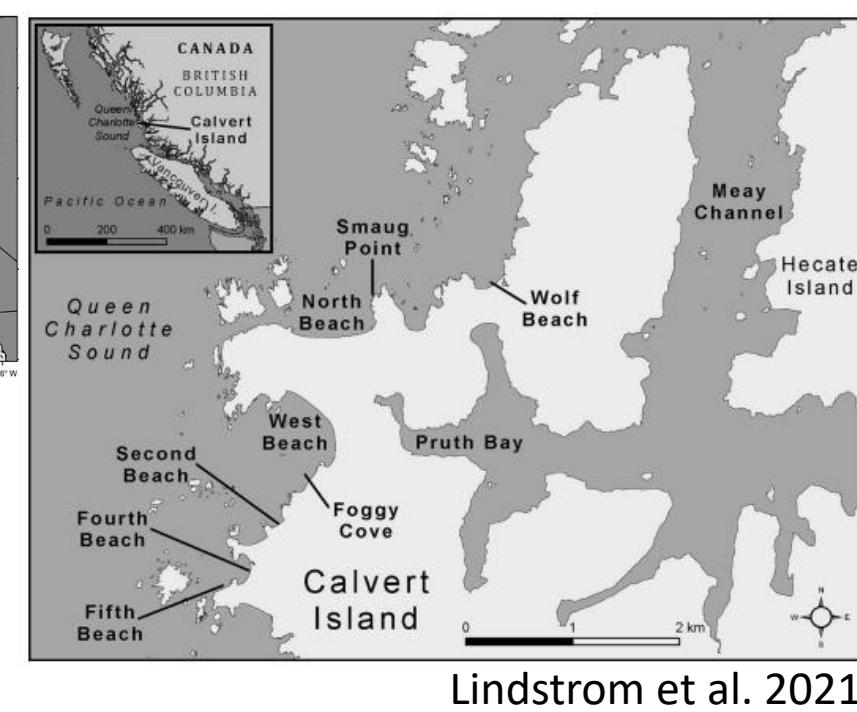
Global



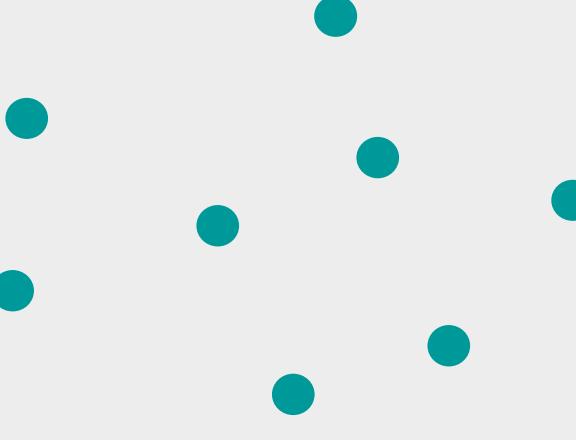
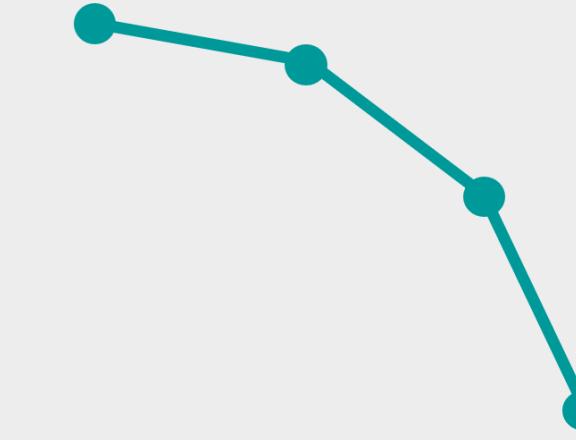
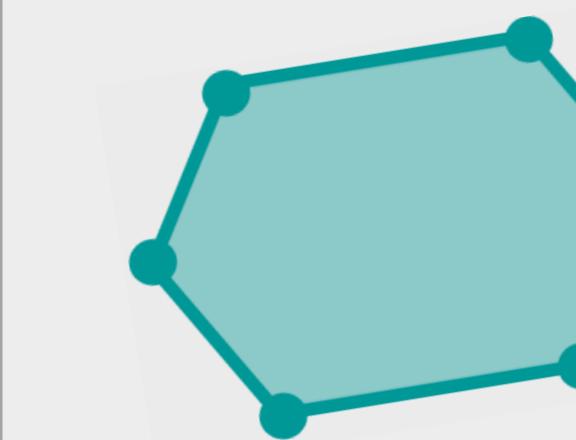
Regional



Local

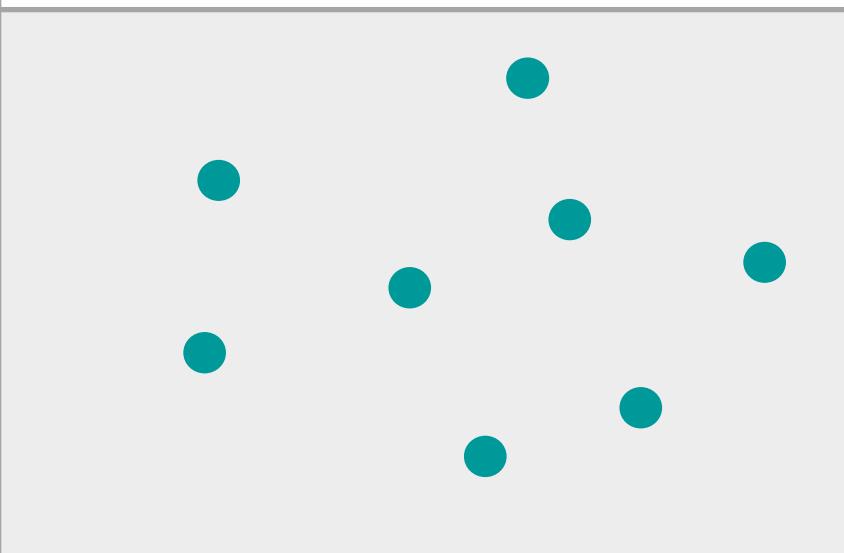


Types of spatial data

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

Types of spatial data

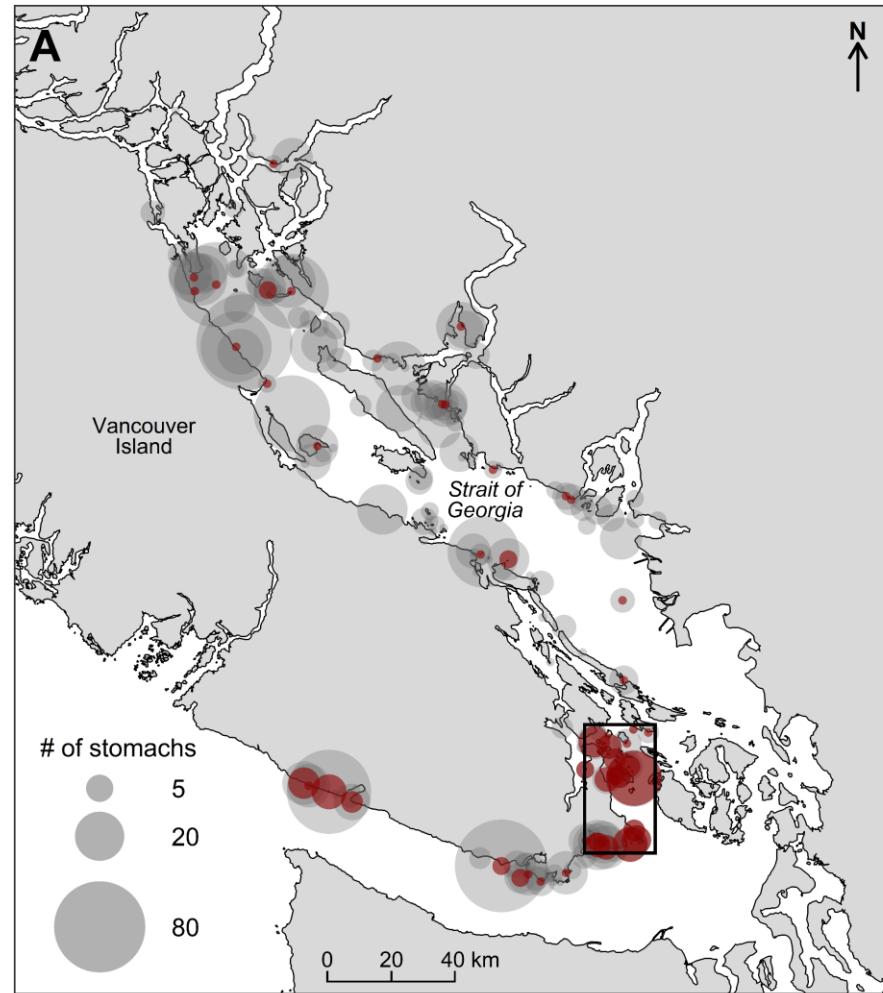
Points



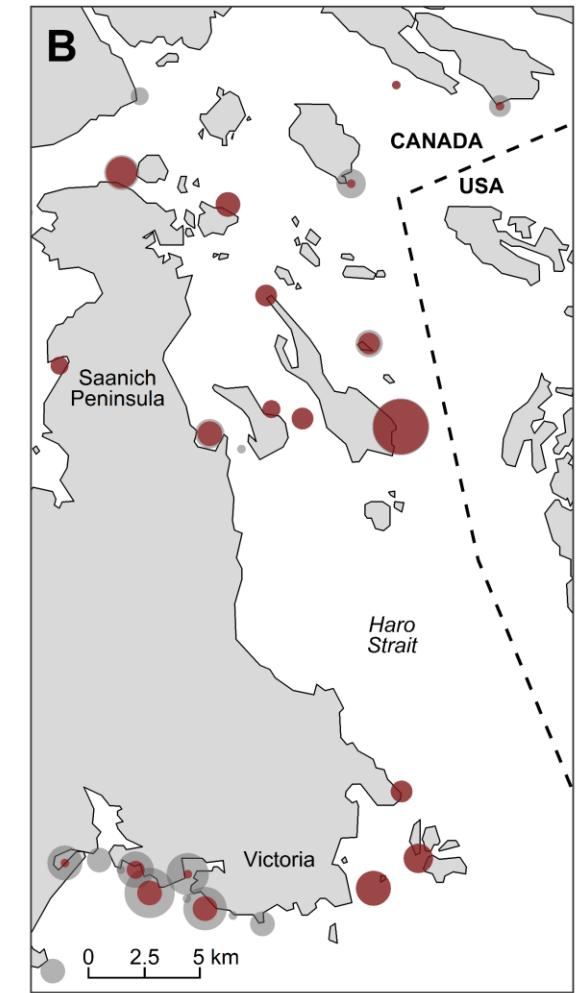
individual x, y positions

examples:

sampling locations
species presence

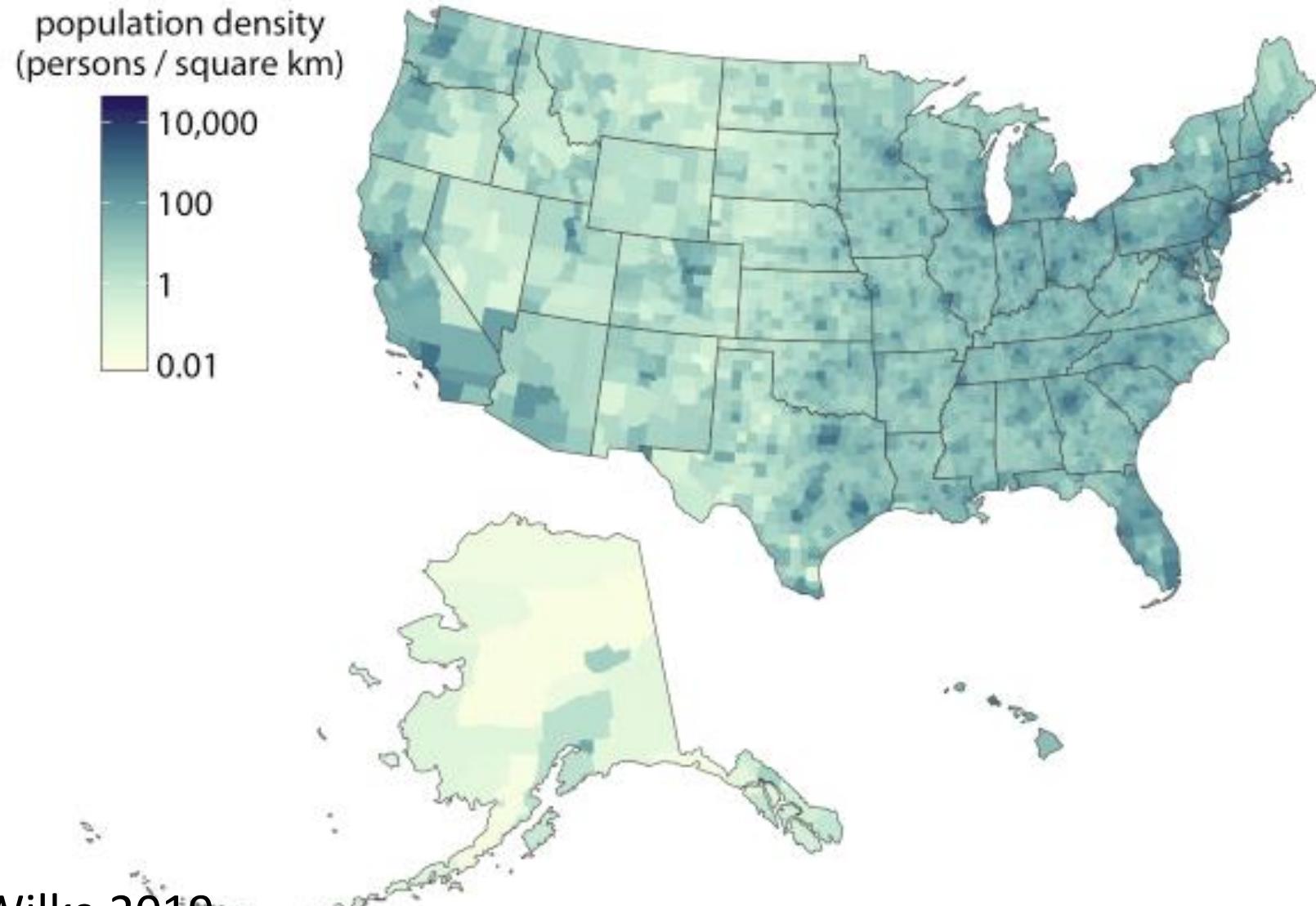


Point size corresponds to number of salmon stomachs containing sand lance

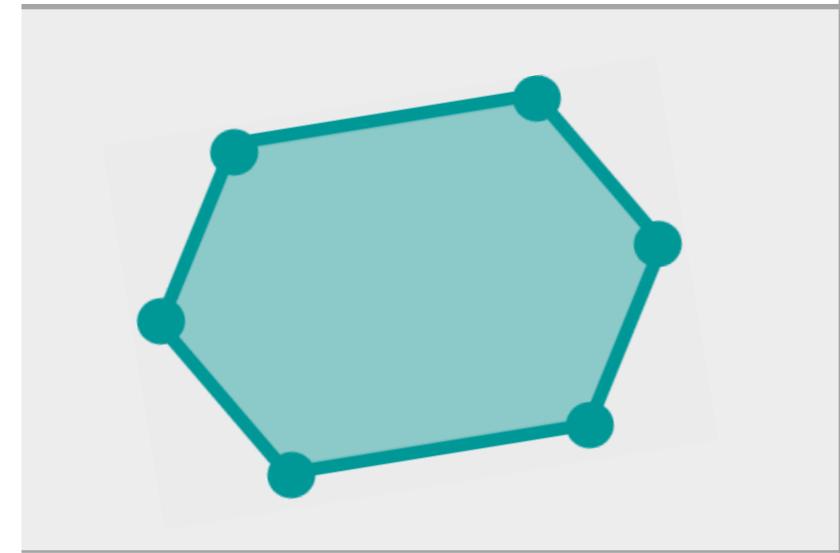


Robinson et al. 2023

Types of spatial data



Polygons



3+ points that connect
and close

coastlines
biogeographic regions

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

Making maps in R:

- **Fits in data processing/analysis scripts**
- Reproducibility (workflow saved in script)
- Nice base maps built in to R packages
- Faceted maps



Why use R for maps?

Load data

```
read_csv()  
importShapefile()  
data(nepacLLhigh)
```

Data wrangling

```
subset()  
merge()  
mutate()
```

Geoprocessing

```
convUL()  
makeGrid()  
projectRaster()
```

All data processing,
analysis, and
visualization steps in
one R script!

Data visualization

```
ggplot() +  
  geom_polygon() +  
  geom_point()
```

Data analysis

```
glm()  
nmds()  
adonis()
```

Which R packages to use?

Main R spatial packages: `sf`, `rnatuRearth` and `PBSmapping`

Available base maps

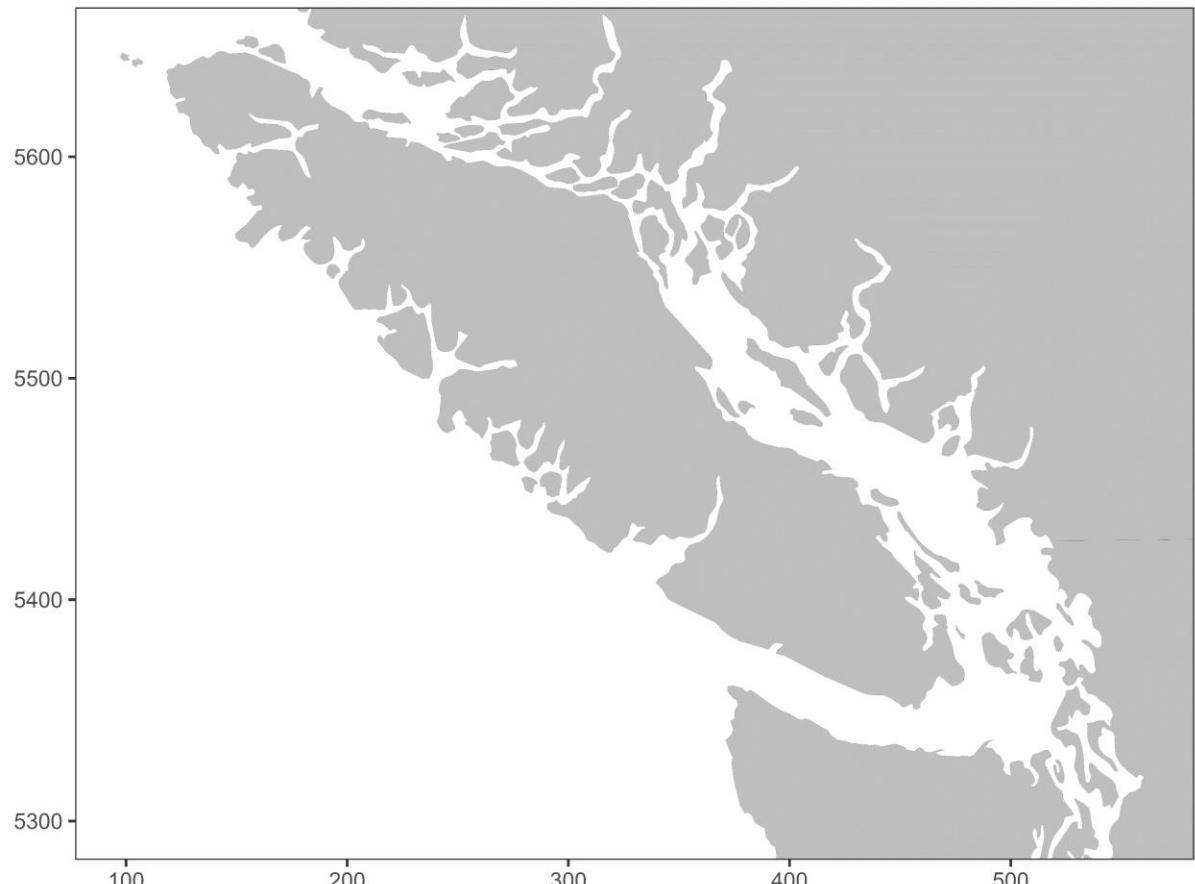
`rnatuRearth::ne_countries` (use with `sf`, `geom_sf()`): coarse resolution, great for global maps, bad for regional/local maps (e.g., Bamfield)

`PBSmapping::nepacLLhigh` (use with `geom_polygon()`): high resolution, great for regional/local maps

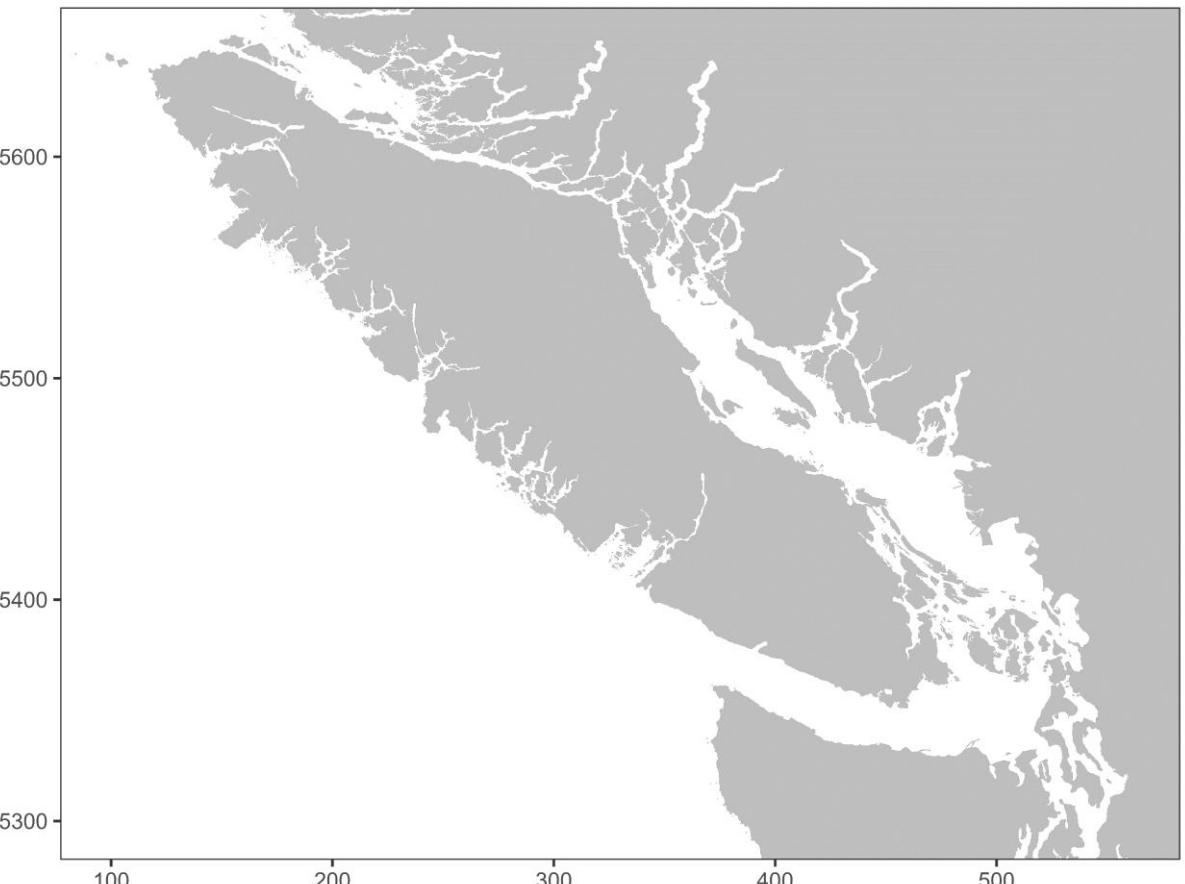
→ We will use PBSmapping basemap and spatial functions

Which R packages to use?

rnatuarlearth

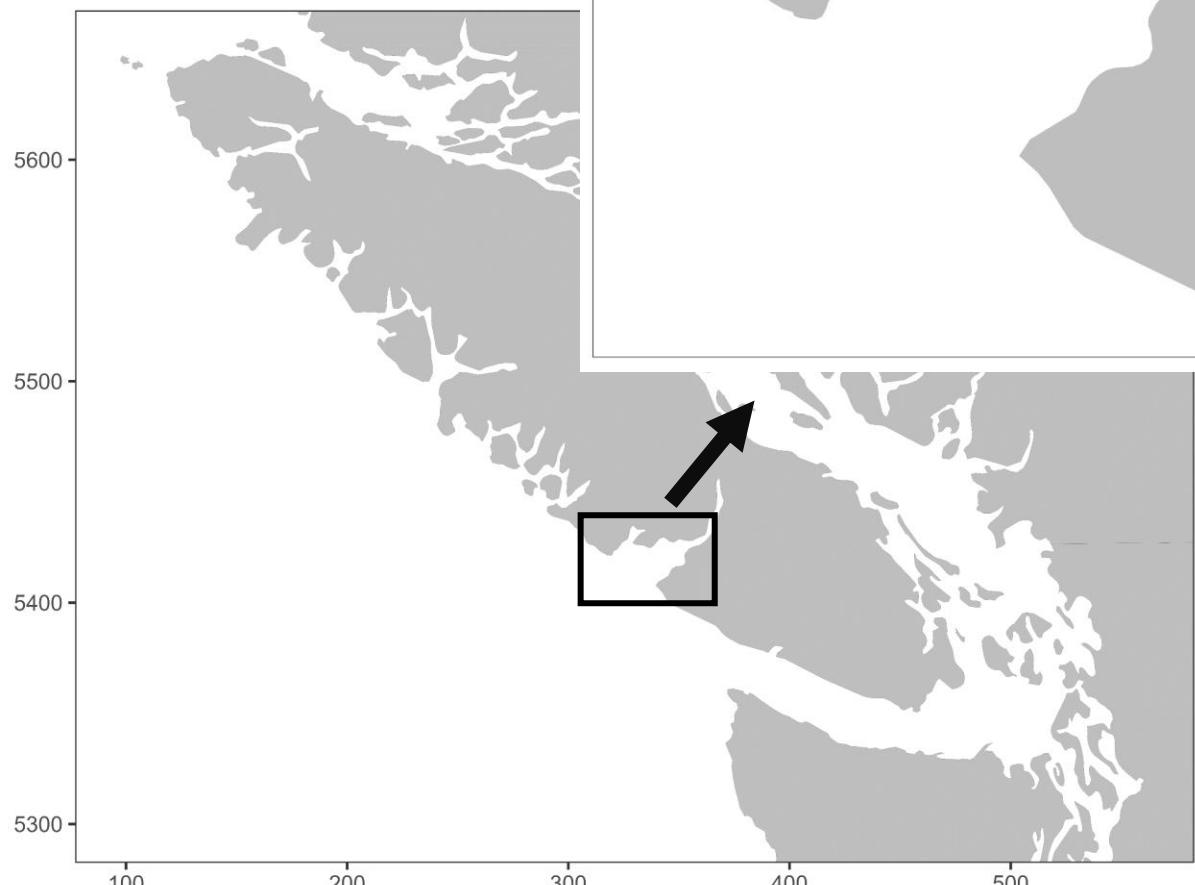


PBSmapping: much more detail

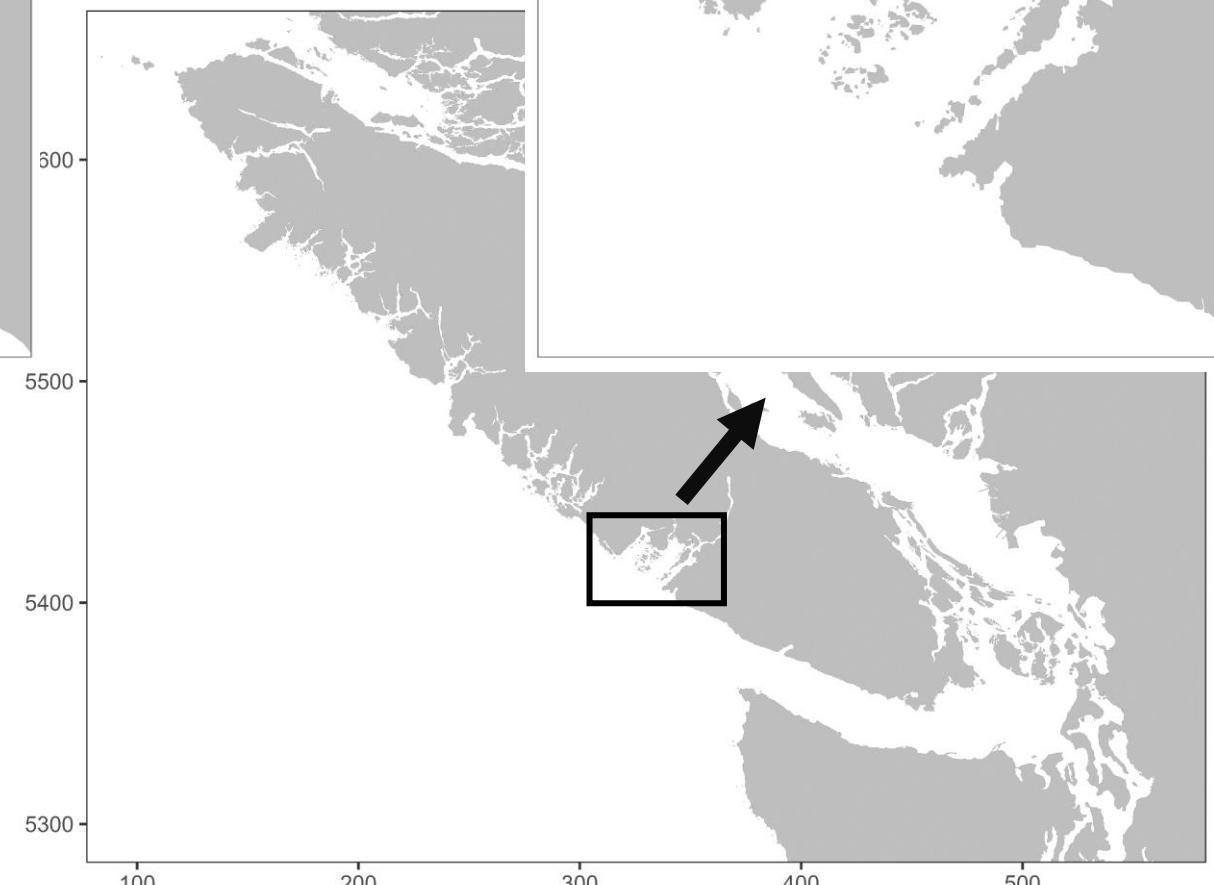


Which R packages to use?

rnatuearth



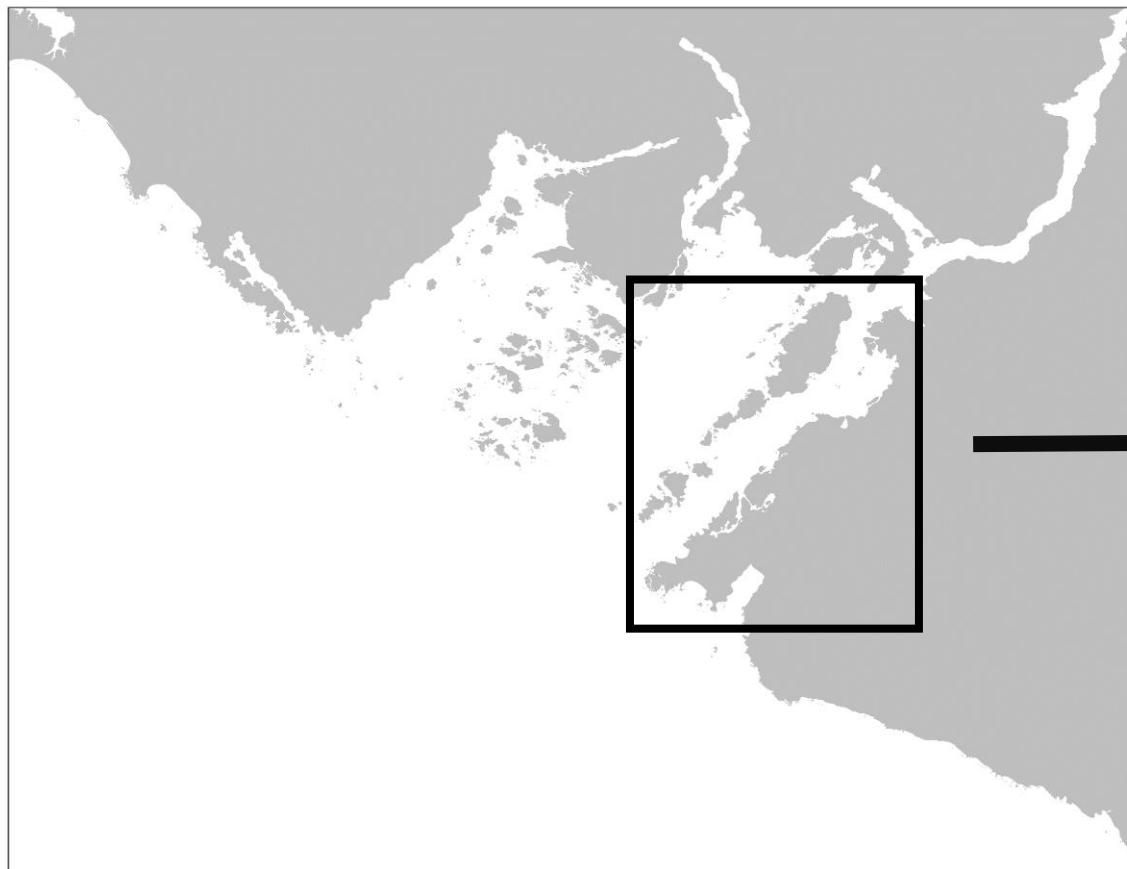
PBSmapping



Zoom in on Barkley
Sound

Which R packages to use?

Even higher resolution is available with shapefiles from Hakai

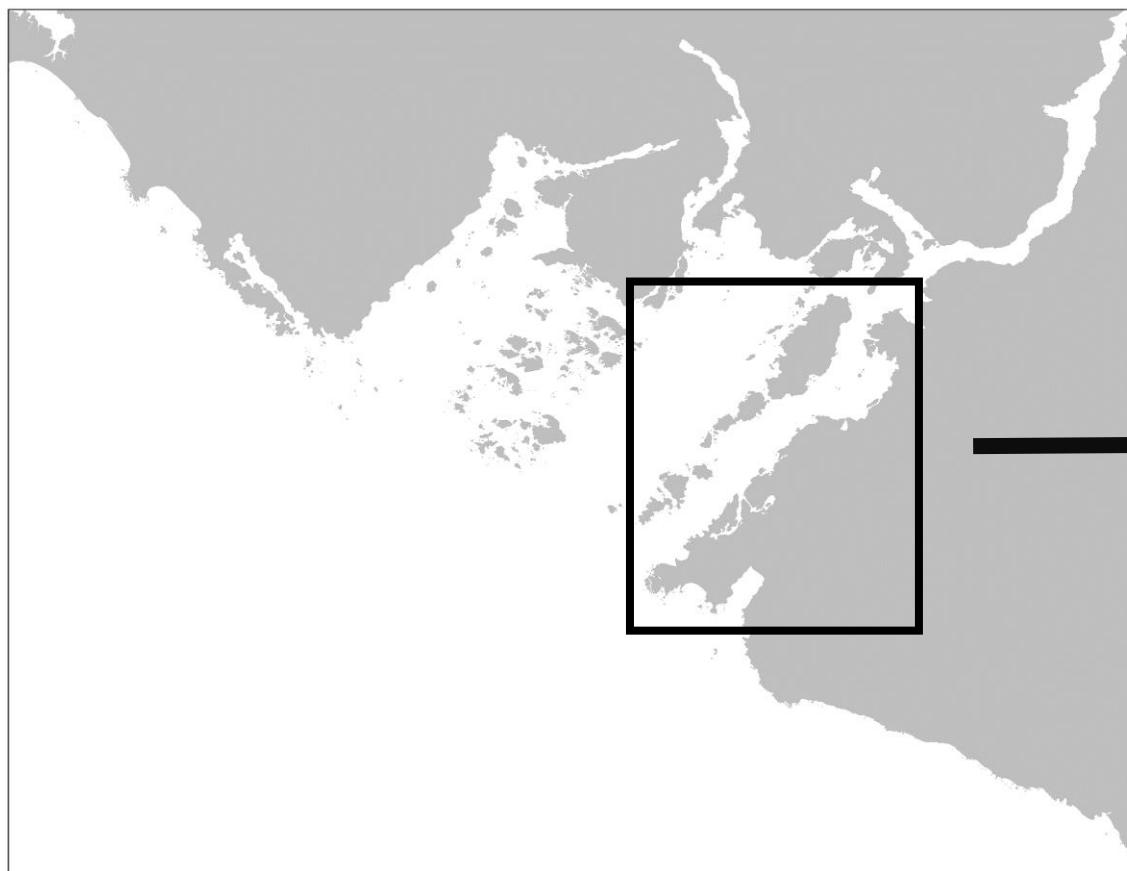


Zoom in



Which R packages to use?

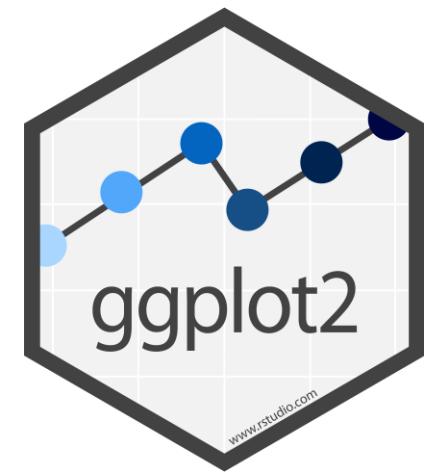
Hakai shapefile is formatted to use same syntax as PBSmapping



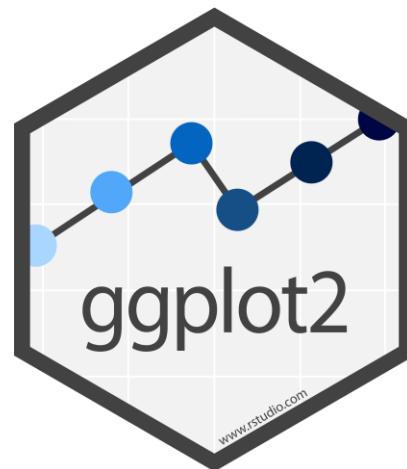
Zoom in



ggplot() “grammar of graphics” in R



ggplot() “grammar of graphics” in R



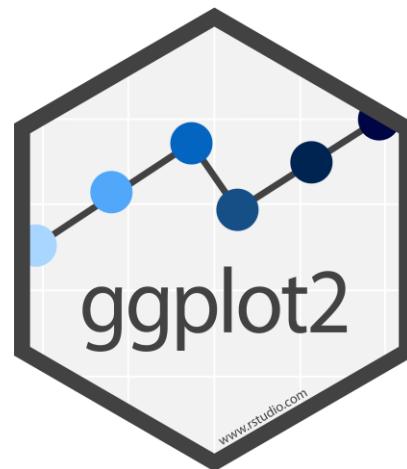
Create figures with multiple layers

Different types of “**geoms**” to make different plots:

- `geom_point()` scatterplots, spatial points
- `geom_line()` line graphs, spatial lines
- `geom_polygon()` 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:

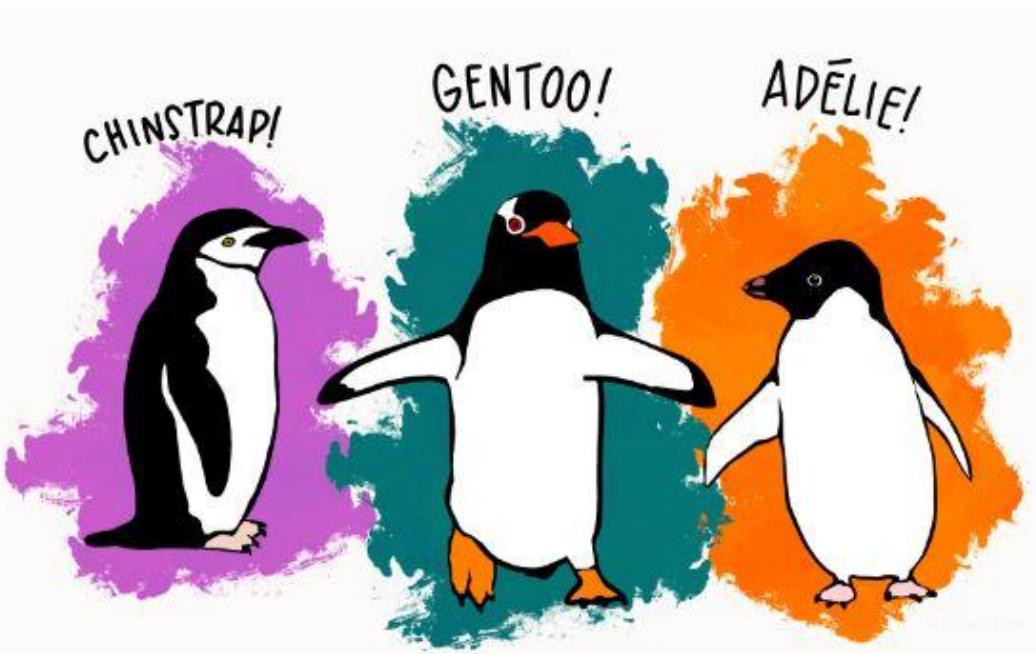
- `geom_point()` scatterplots, spatial points
- `geom_line()` line graphs, spatial lines
- `geom_polygon()` shapefiles (spatial polygons)
- `geom_label()` print text labels on figure

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

The code snippet shows the `ggplot()` function followed by a plus sign, then `geom_point()`. The `data` argument is highlighted with a yellow background and has an arrow pointing to it from the word **data**. The `aes()` argument is highlighted with a green background and has an arrow pointing to it from the word **aesthetics**.

ggplot() example

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



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



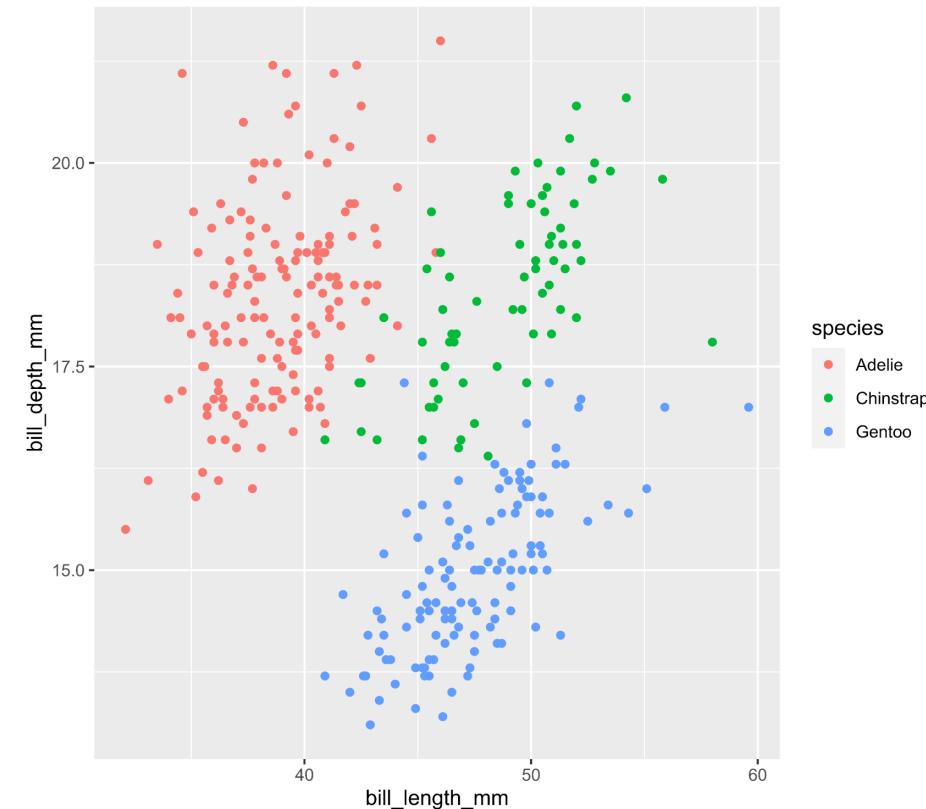
eBird



Australian Museum

ggplot() example

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

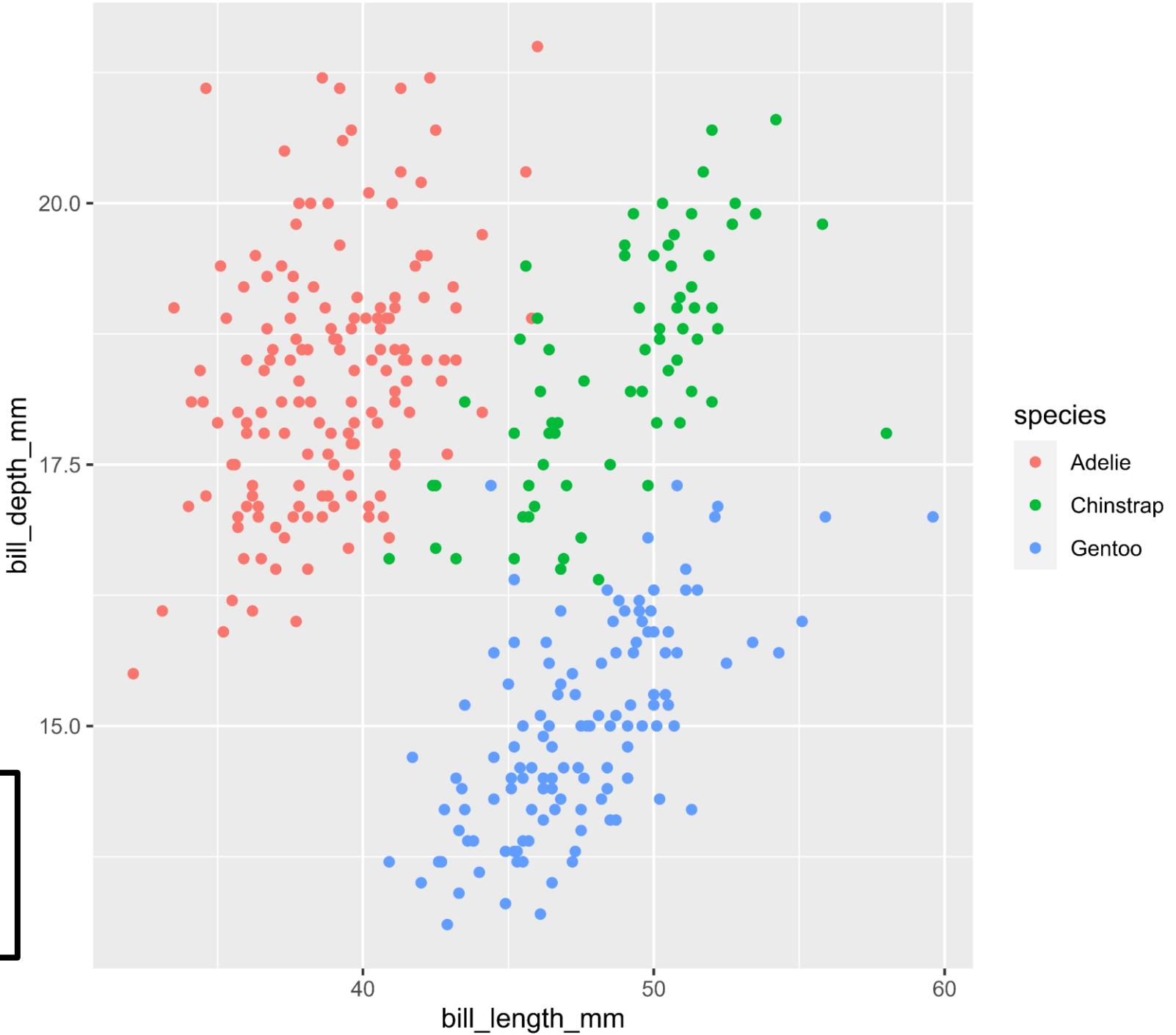


```
data(penguins)

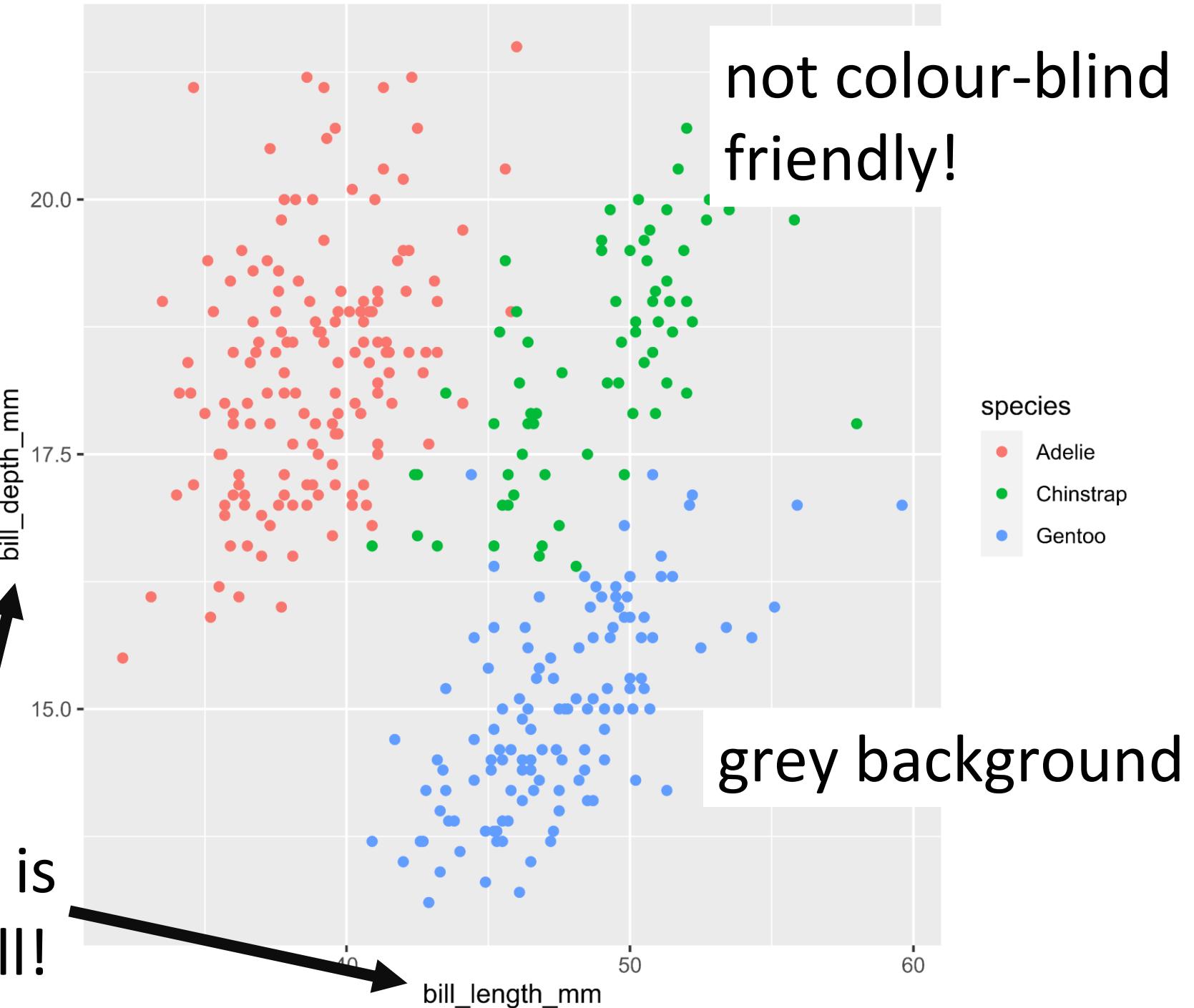
ggplot() +
  geom_point(data = penguins,
             aes(x = bill_length_mm, y = bill_depth_mm, color = species))
```

**DO NOT USE
ggplot()
defaults!**

What do you think
should be changed?



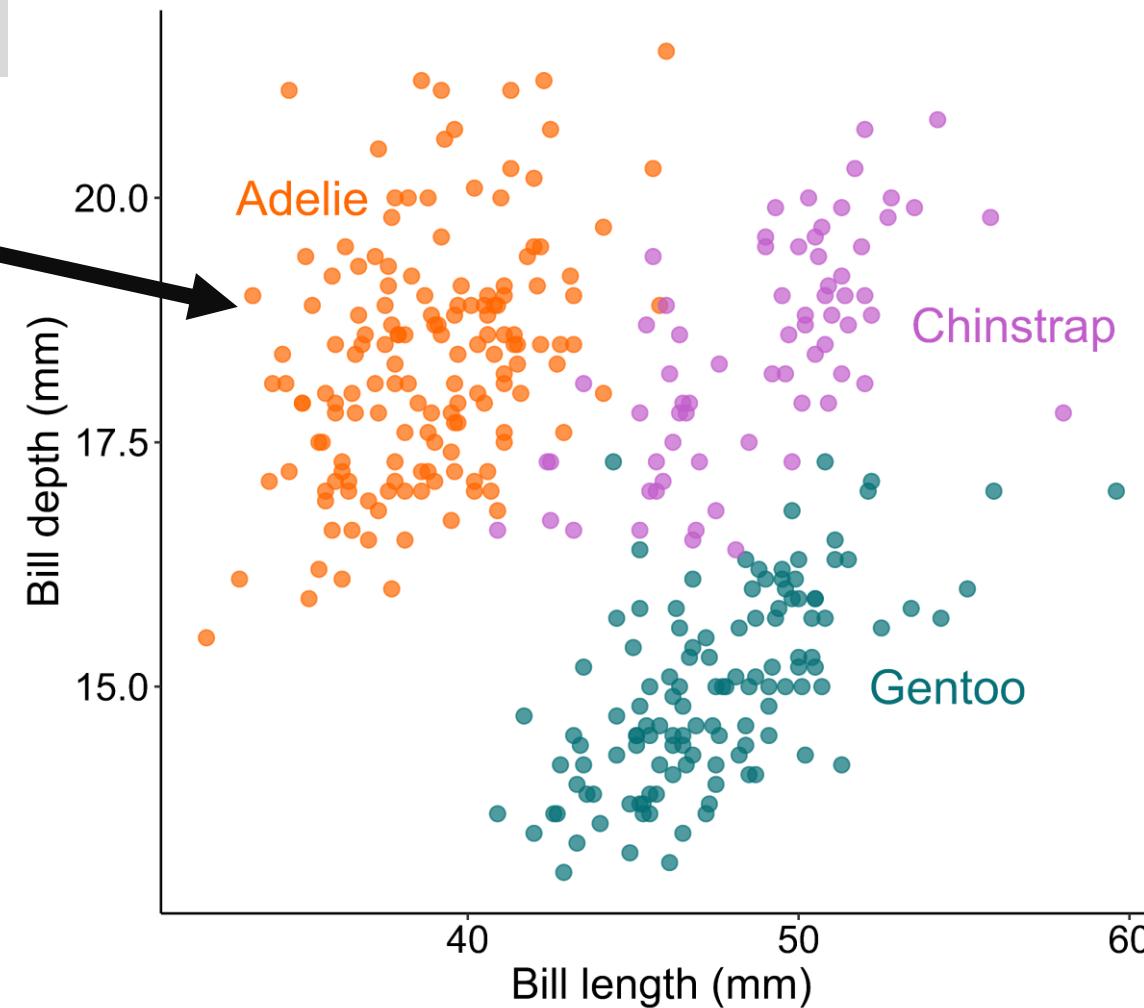
**DO NOT USE
ggplot()
defaults!**



```
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

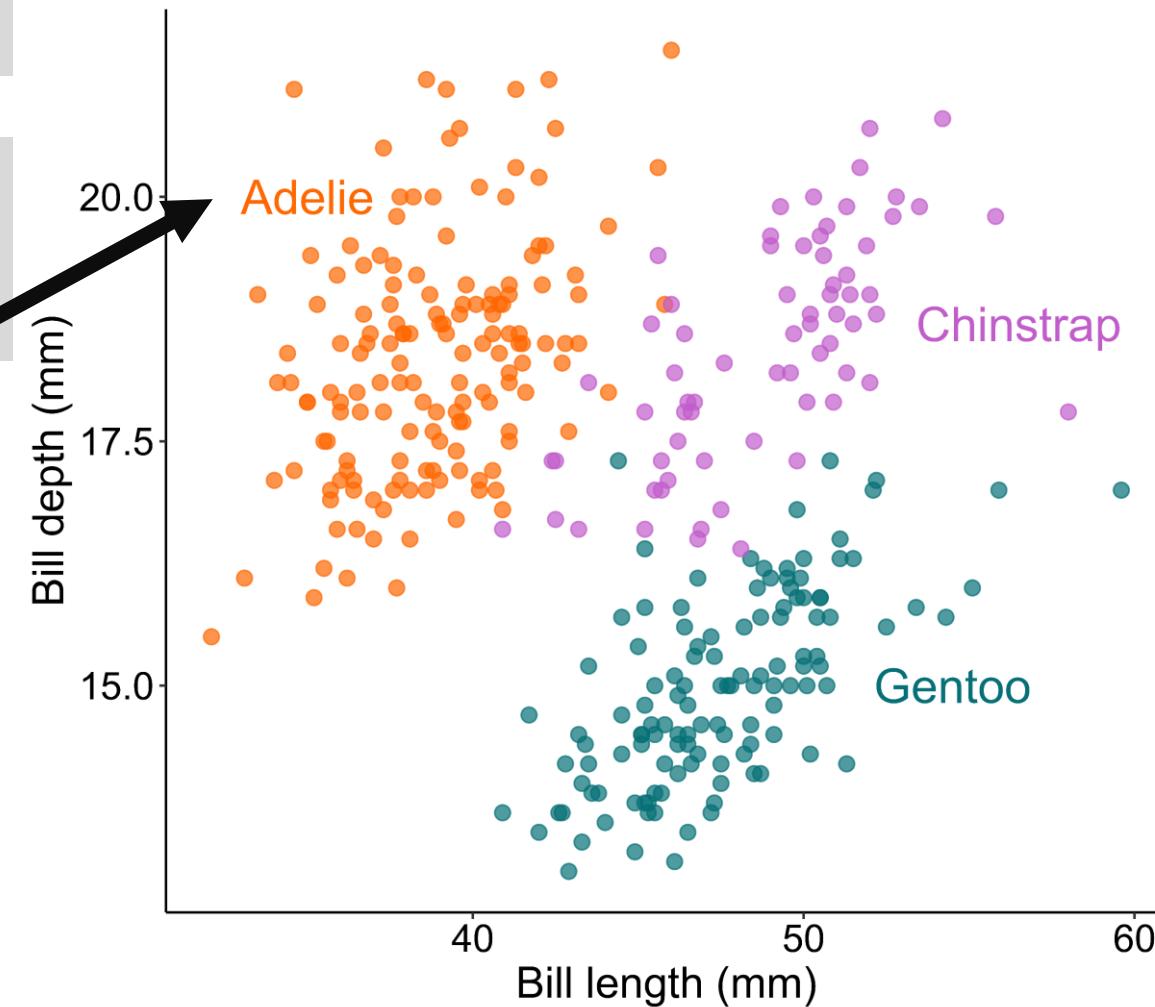


```
ggplot() +  
  geom_point(data = penguins,  
             aes(x = bill_length_mm, y = bill_depth_mm, color = species))
```

```
  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))
```

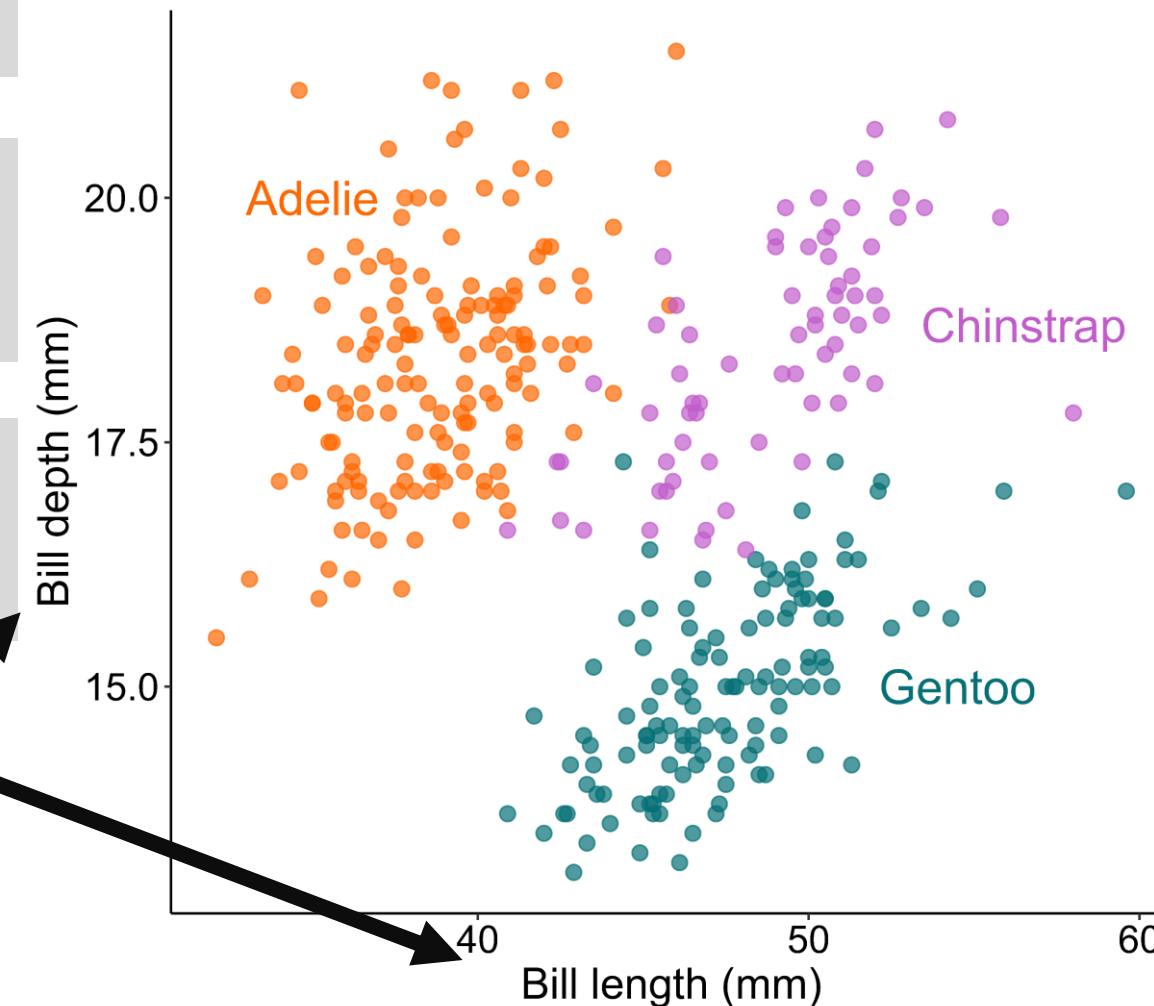
```
  geom_point(size = 2.7, alpha = 0.7)
```

```
  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))
```

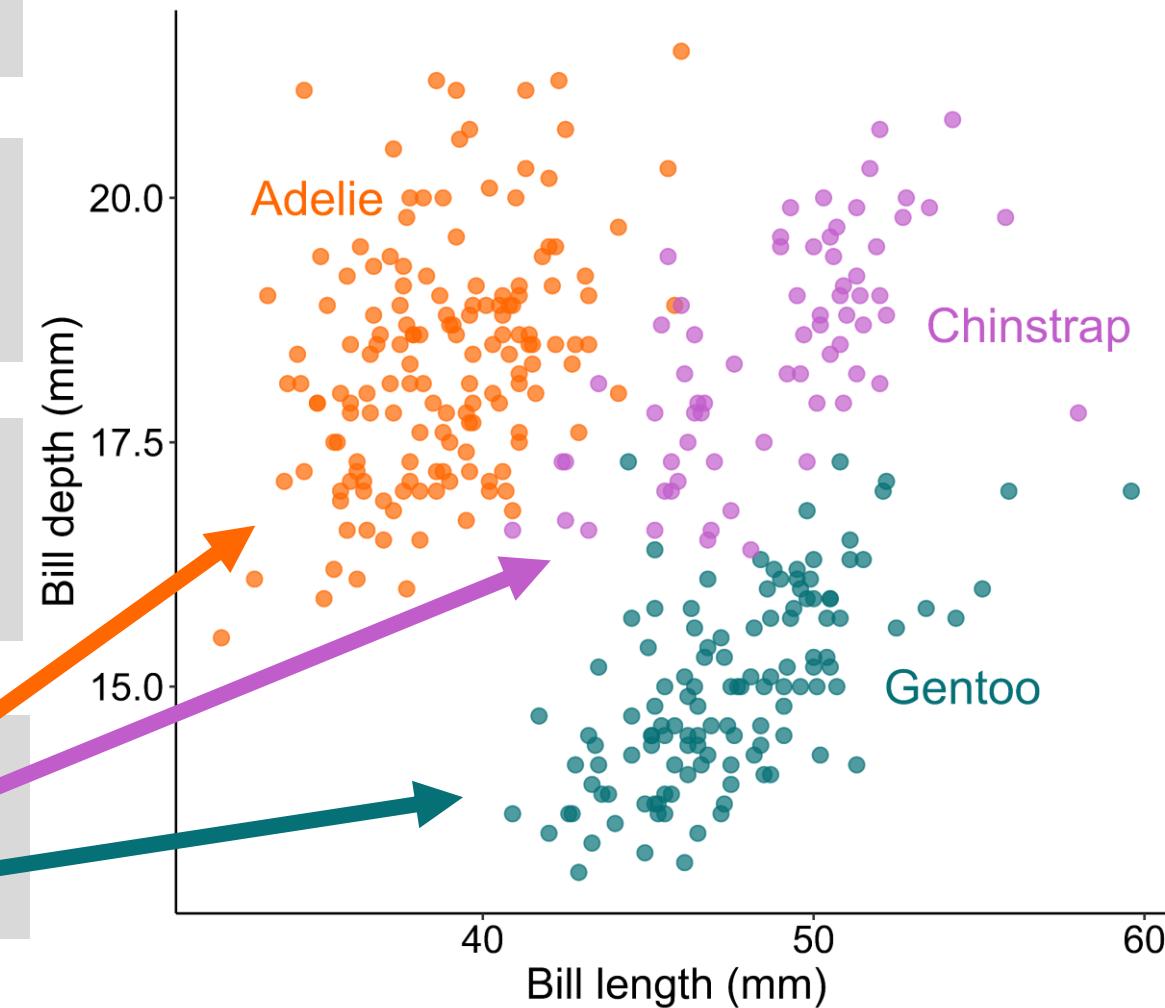
```
  geom_point(size = 2.7, alpha = 0.7)
```

```
  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

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

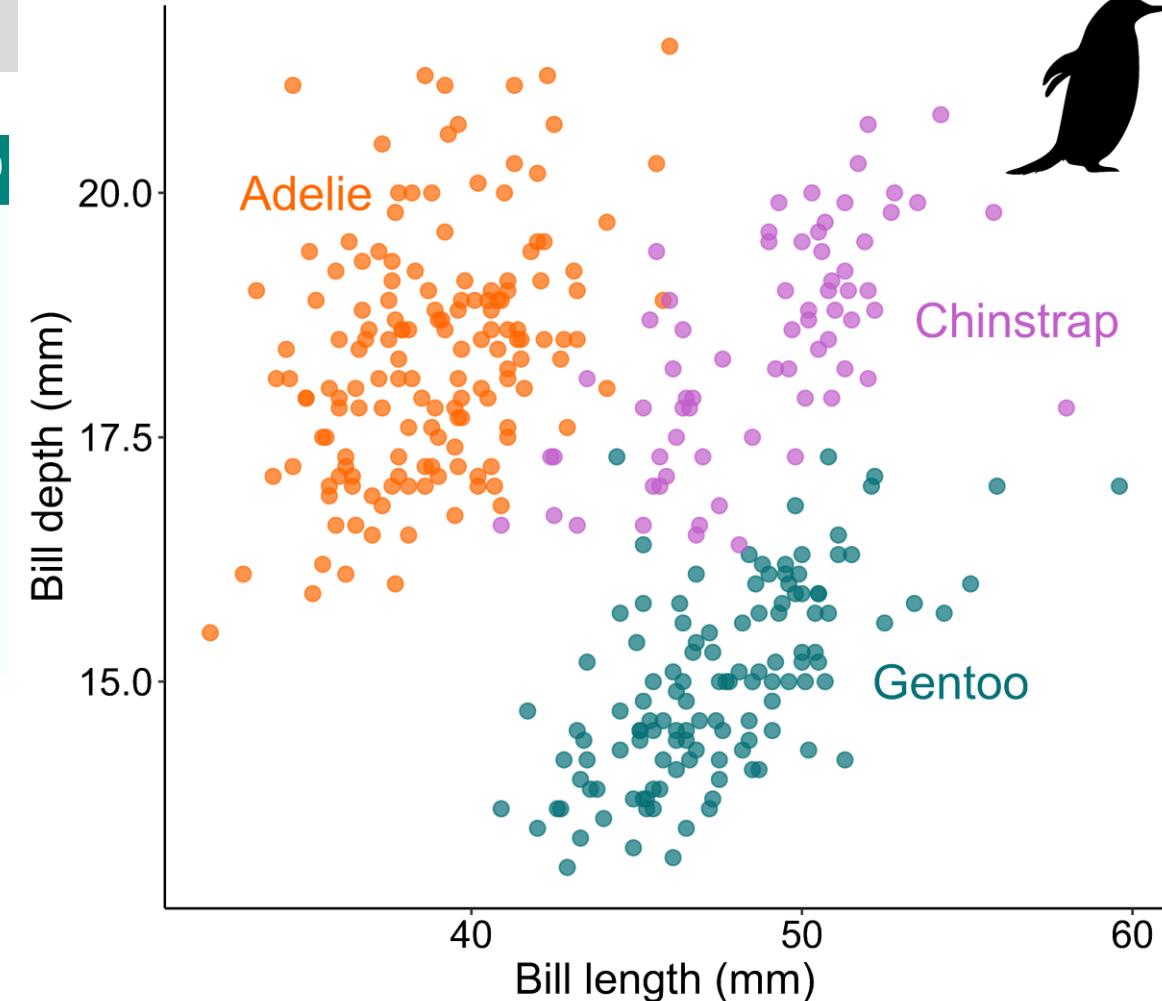
Gento penguin scientific name

PHYLOPic Enter the name of a group of organisms.

Free silhouette images of animals, plants, and other life forms, available for reuse under Creative Commons licenses.

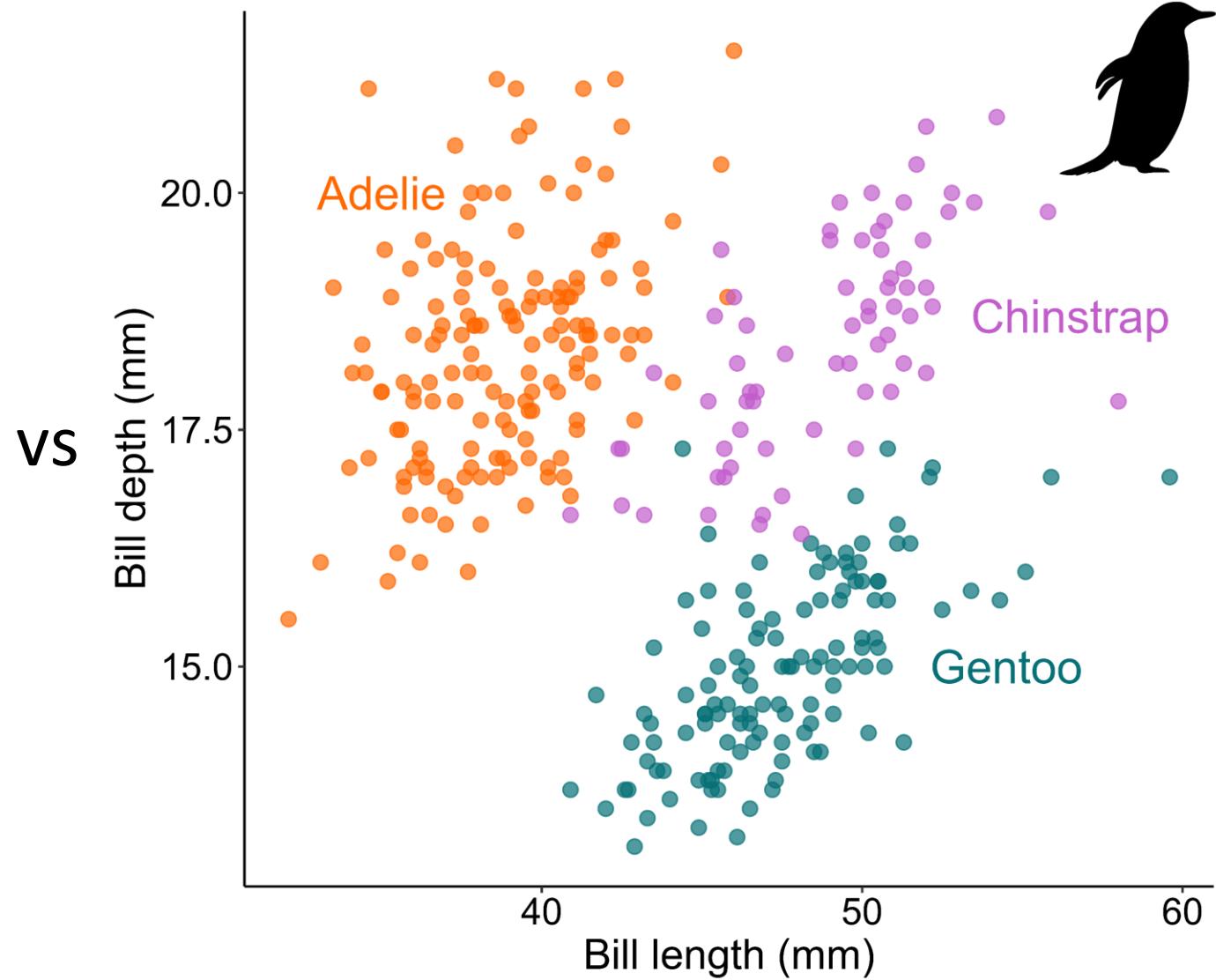
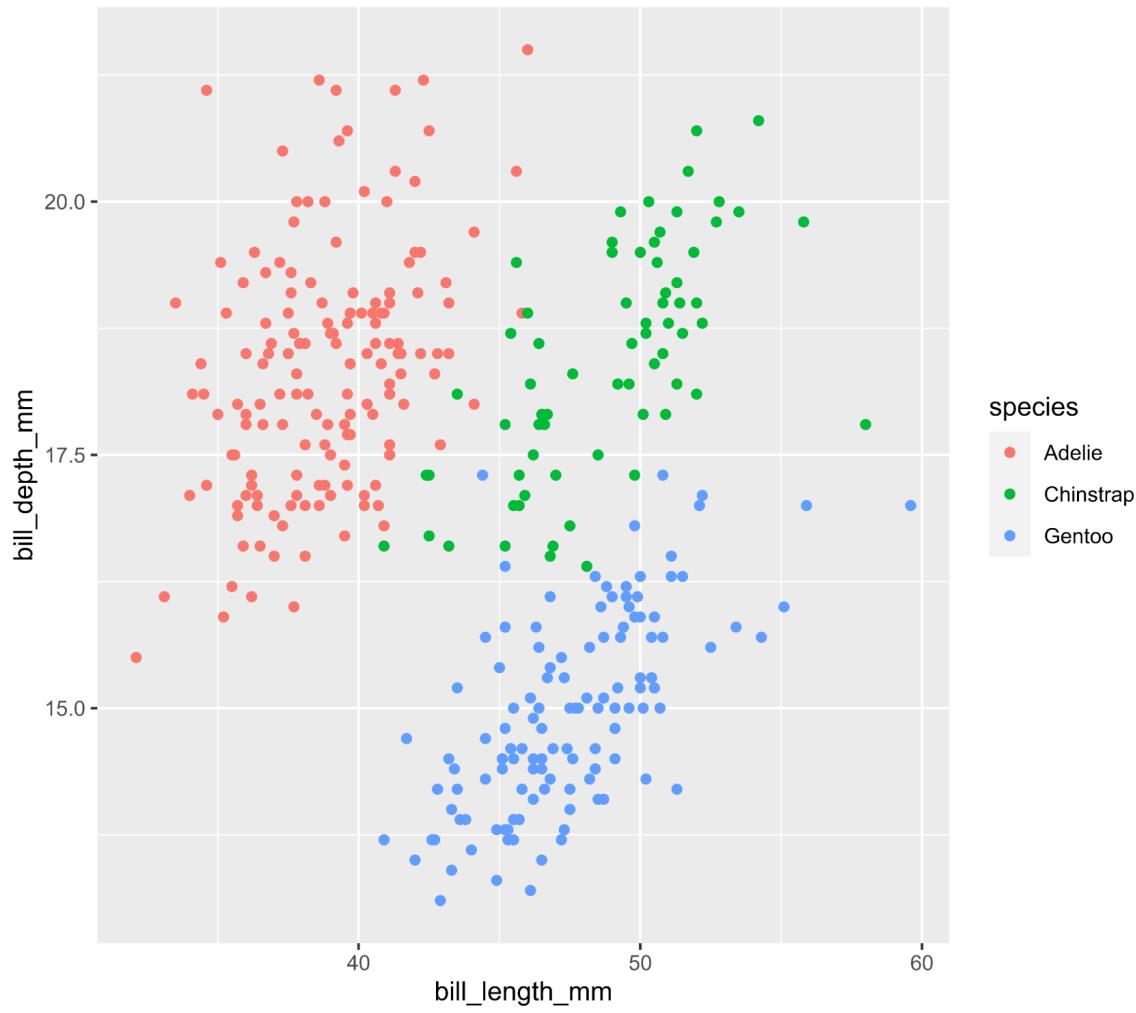
Latest Uploads

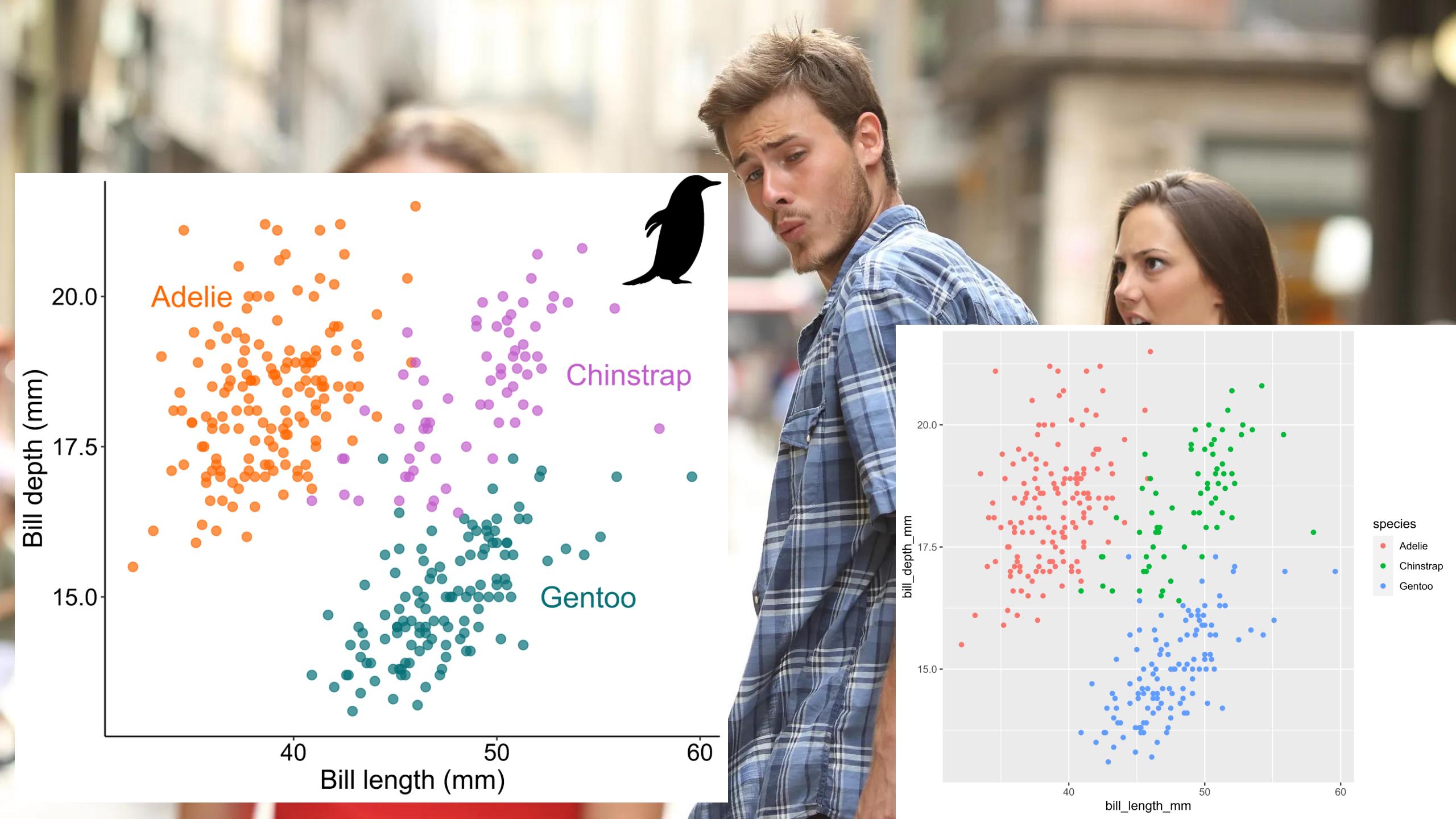
SEE MORE →



<https://www.phylopic.org/>

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





ggplot() summary

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

Do not use ggplot() defaults!

Have fun/be creative with ggplot()!

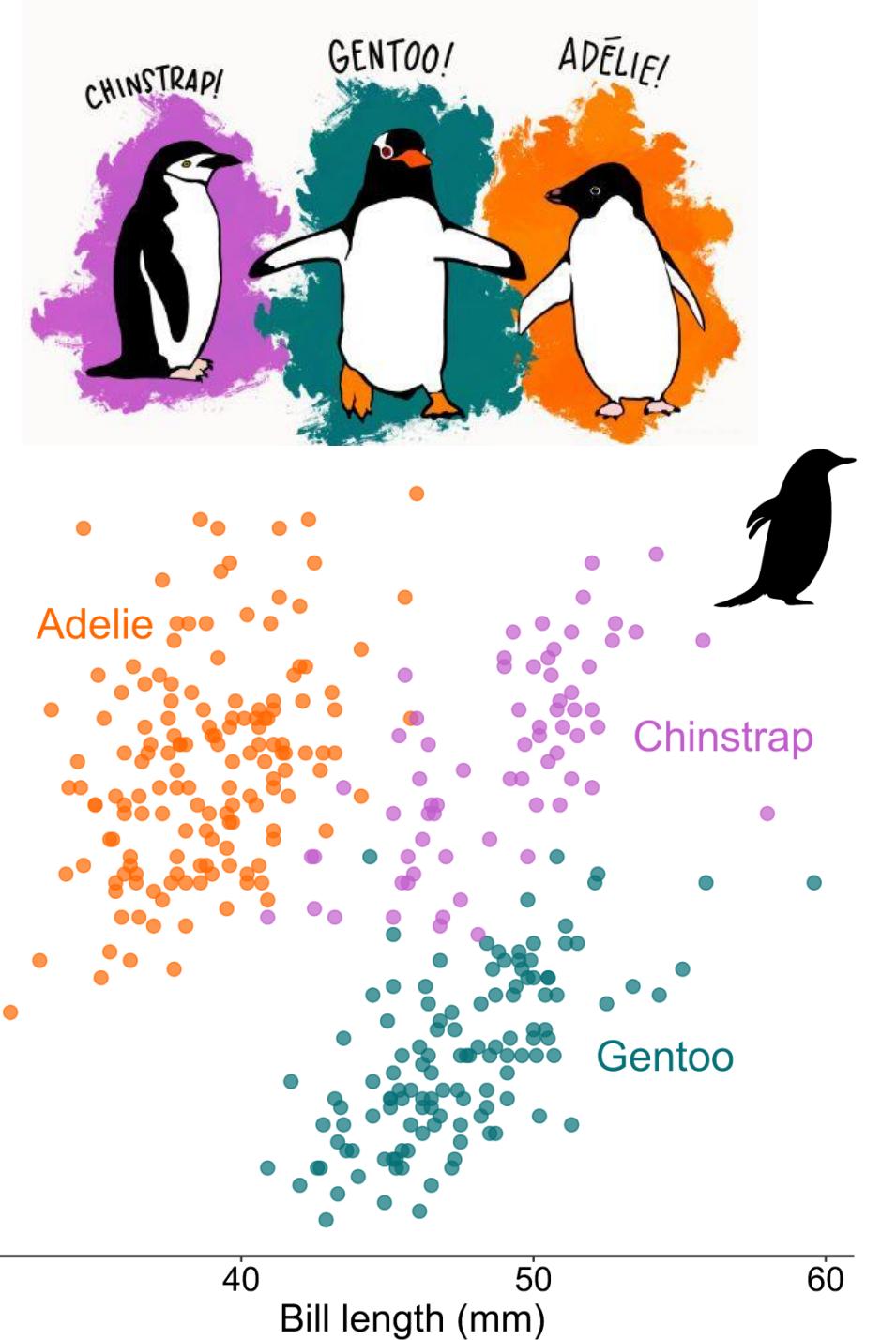
Additional resources

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/>

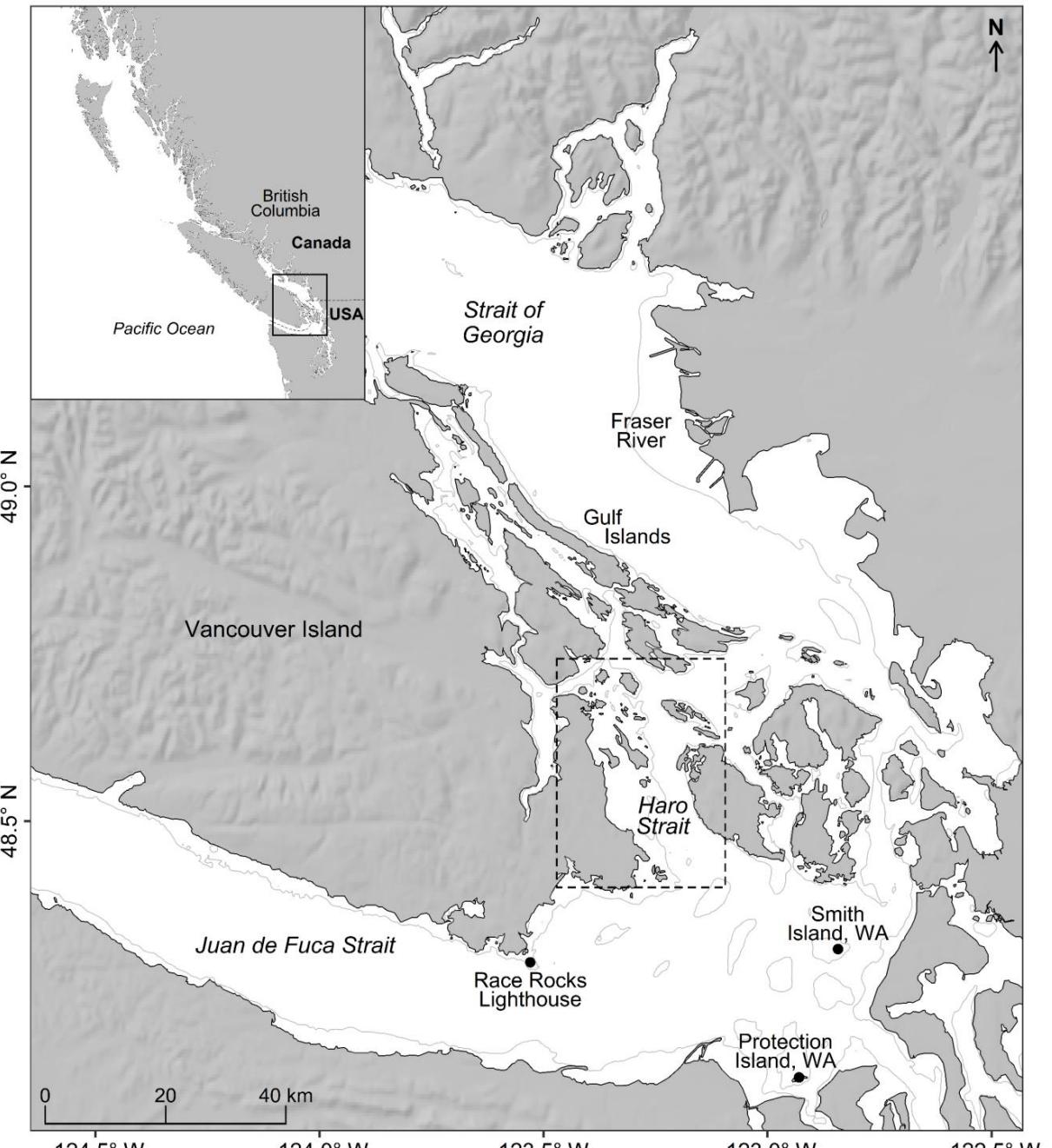


Study area maps

Goal: show sampling locations and important geographic/biological features

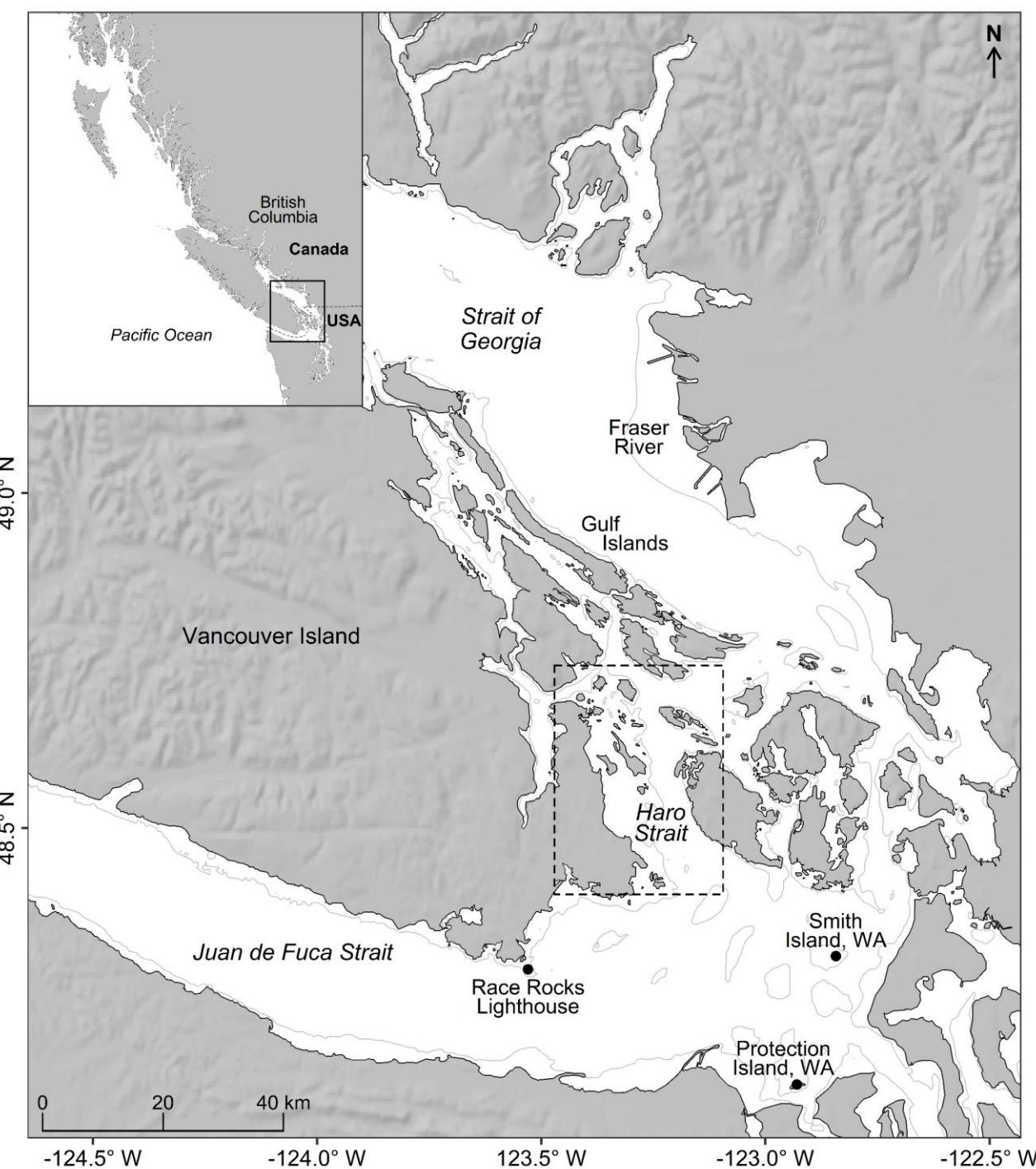
Often greyscale for simplicity

Include inset map showing larger geographic region



Important parts

**Good spatial
resolution is key**

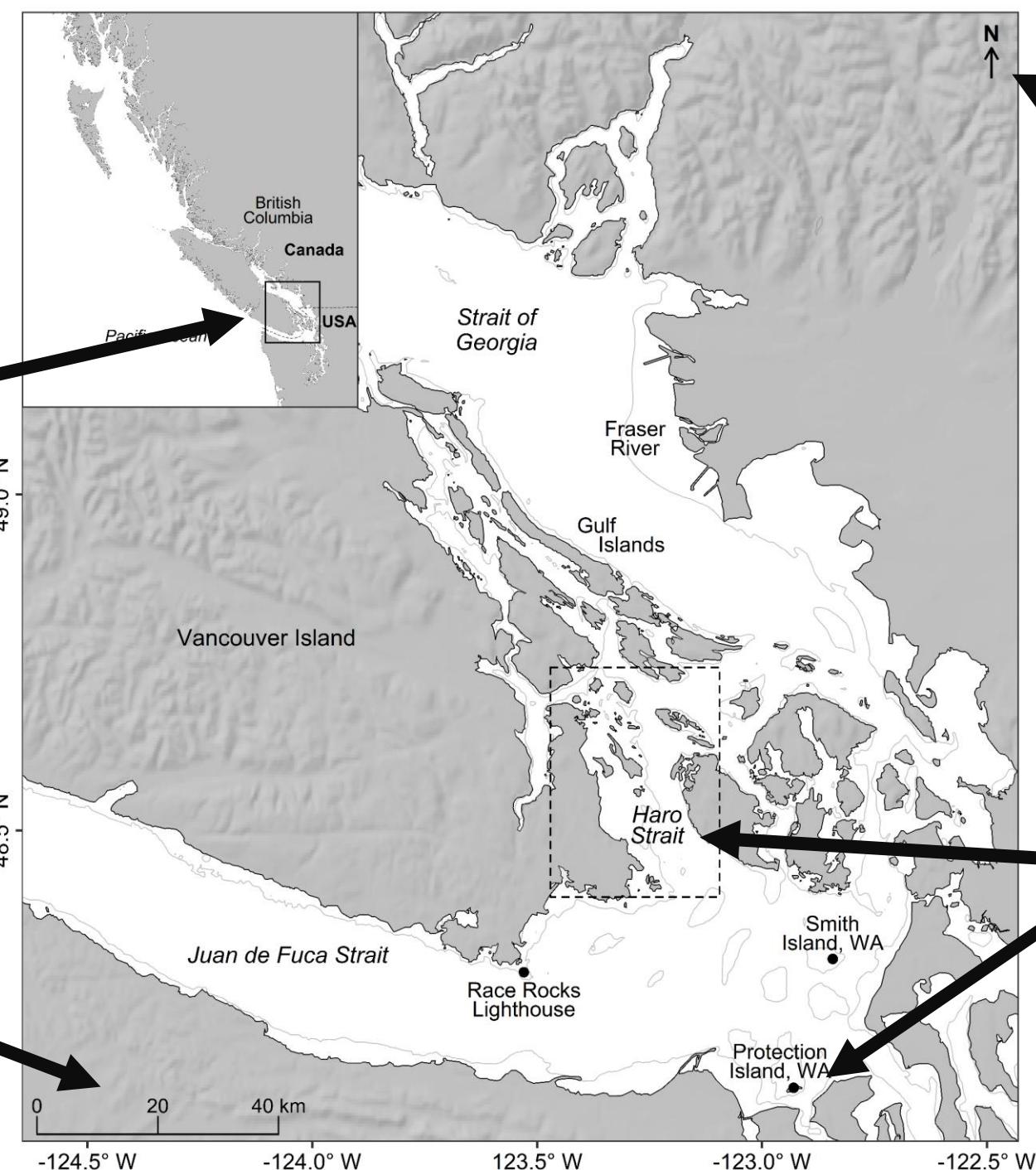


Important parts

Inset map showing where the study area is

Study area denoted by rectangle

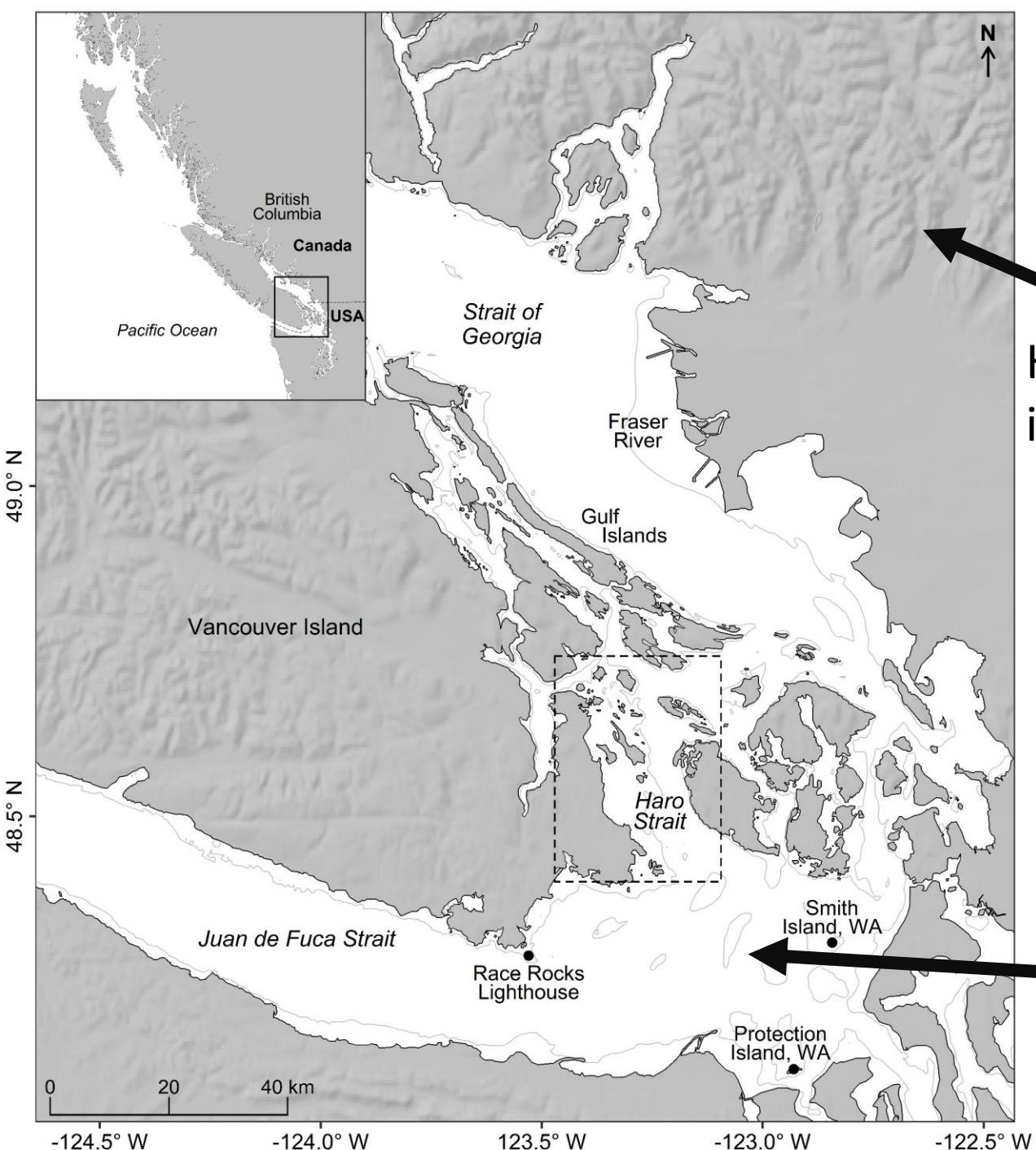
Scale bar



North arrow

Important sites labelled

Extra pieces



Hillshading to illustrate topography

50 metre depth contours

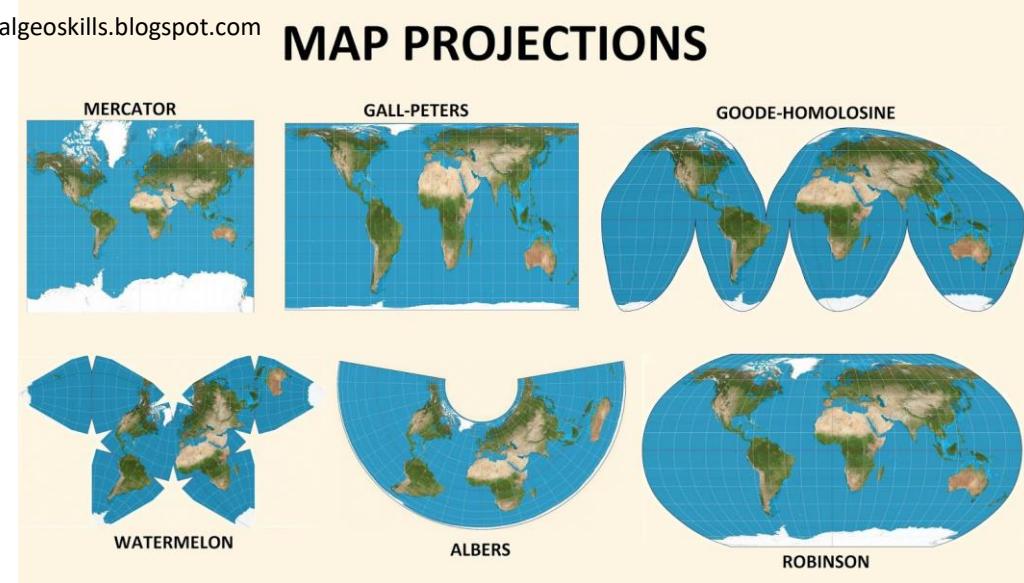
Simple study area map

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)



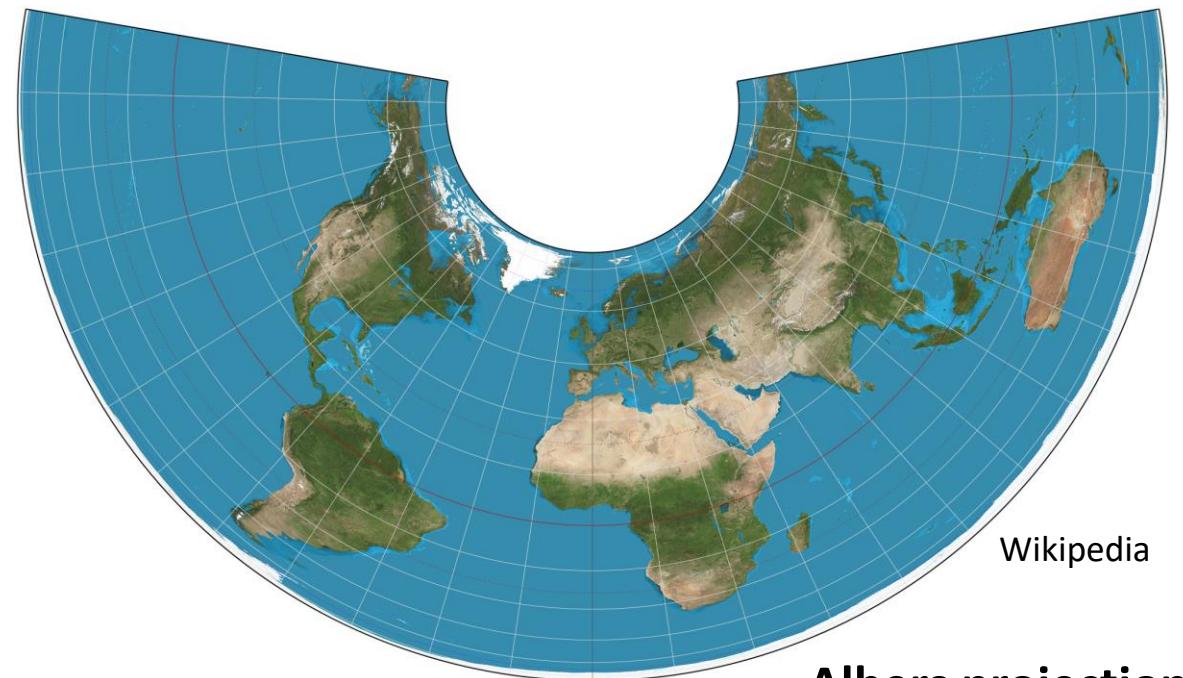
Simple study area map

What projection to use?

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

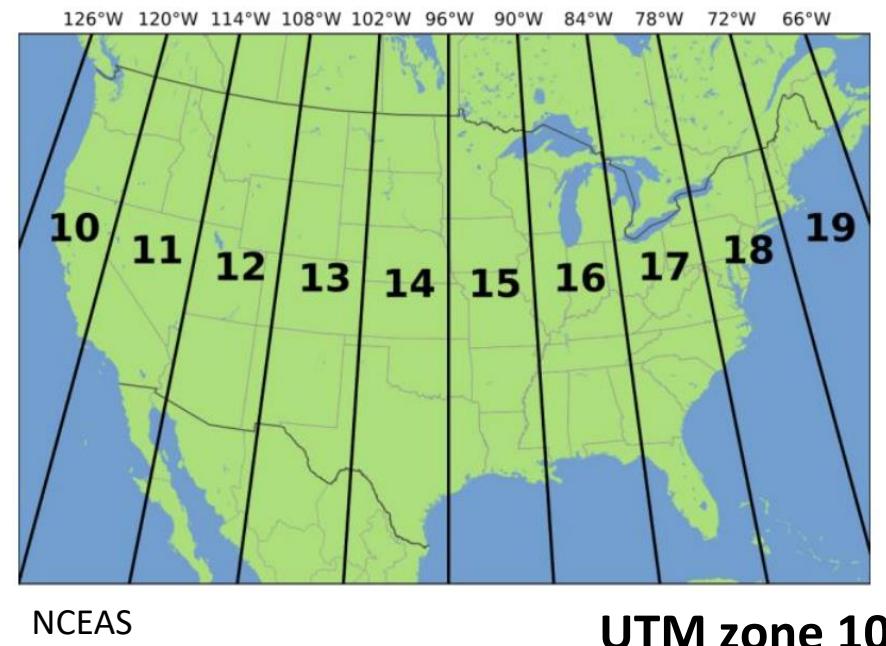
In BC, we want to use **BC Albers** or **UTM zone 9 or 10**

Both use UTM Northings and Eastings (in metres/kilometres) instead of latitude and longitude (in degrees)



Wikipedia

Albers projection



NCEAS

UTM zone 10

Simple study area map

World Mercator projection with true country size and shape added

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 or UTM zone 9 or 10**

Both use UTM Northings and Eastings (in metres/kilometres) instead of latitude and longitude (in degrees)



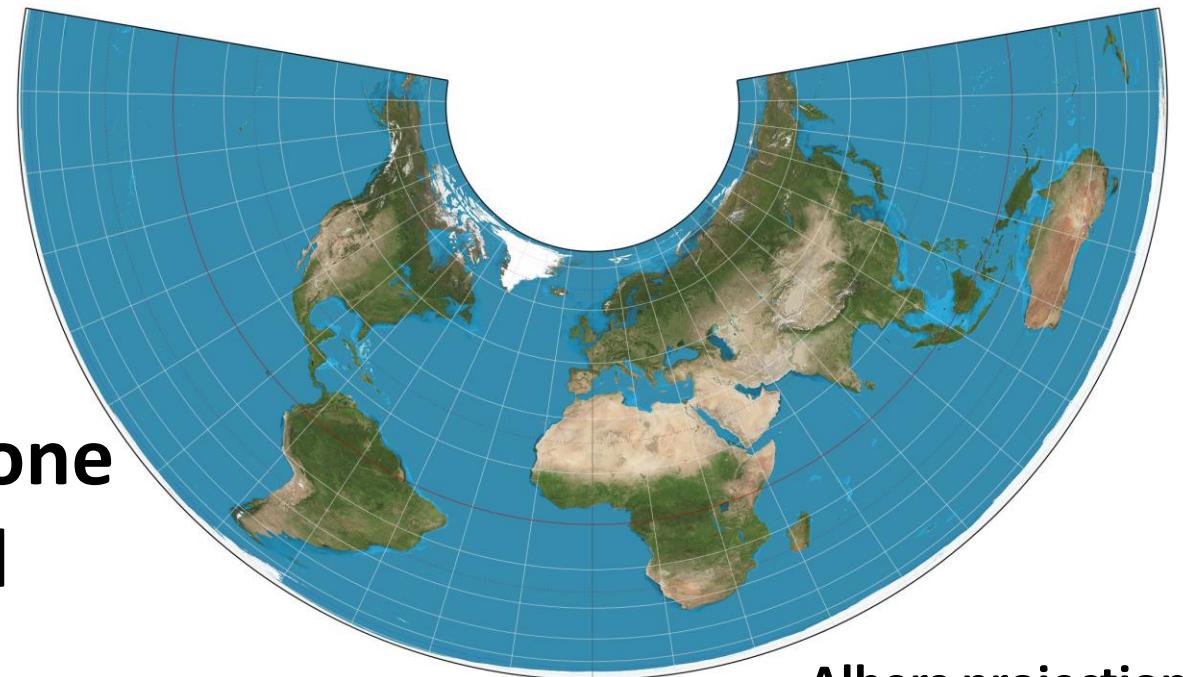
@neilrkaye

Simple study area map

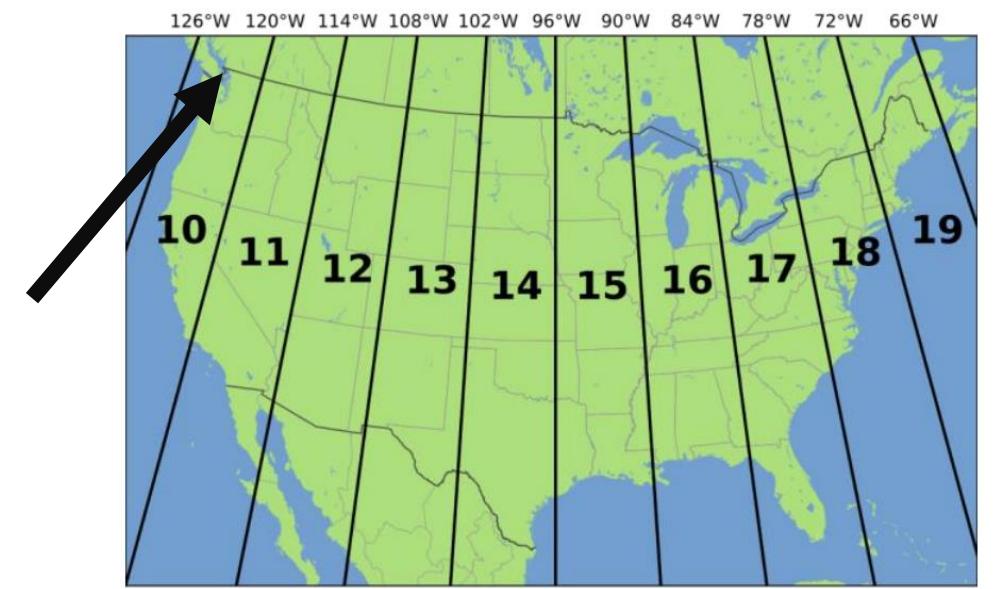
What projection to use?

In this workshop, we will use **UTM zone 10** because PBSmapping has a useful function for converting latitude/longitude (WGS 84 projection) to UTM zone 10

PBSmapping::convUL()

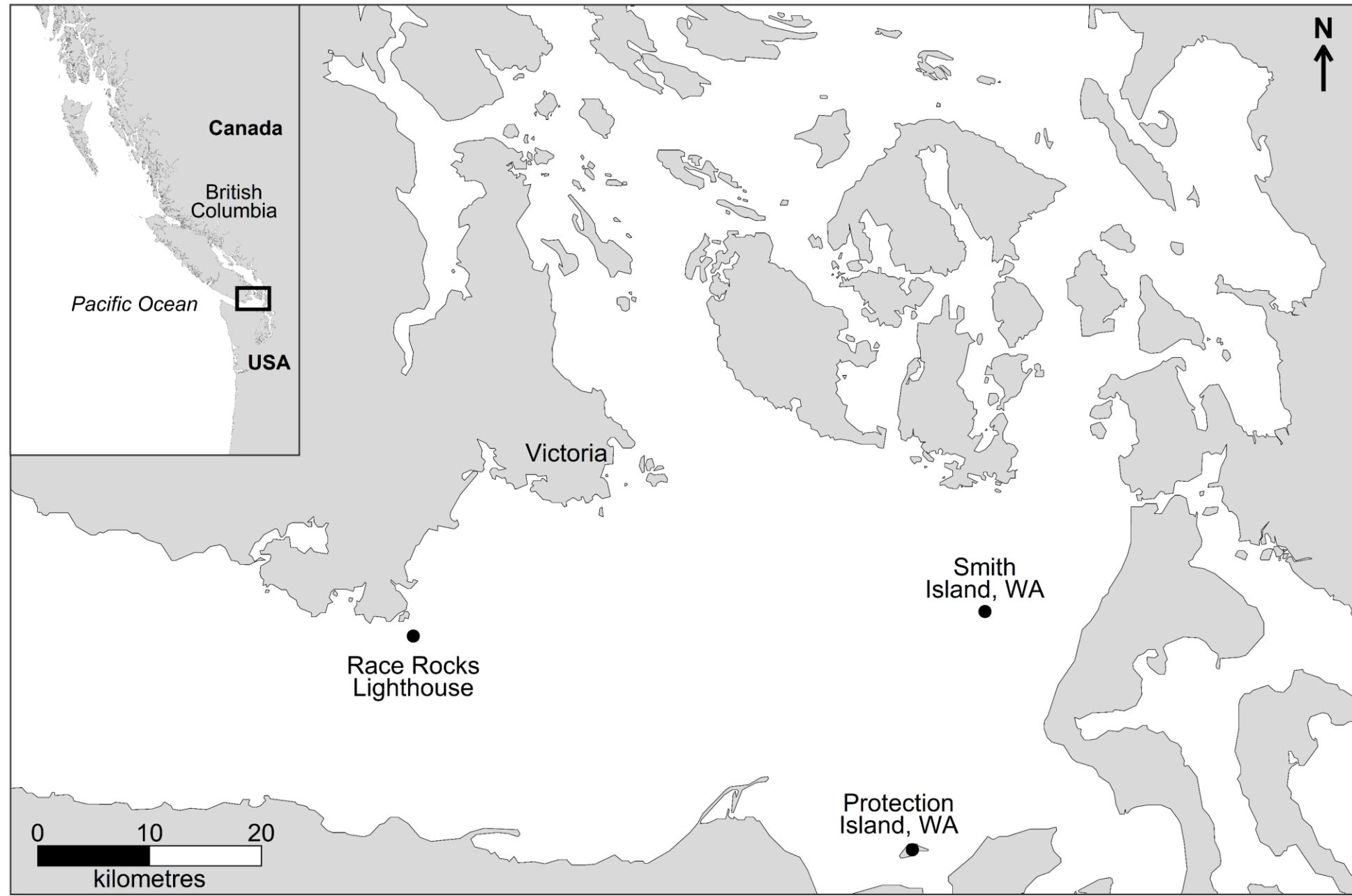


Albers projection



UTM zone 10

Simple study area map



Load nepacLLhigh basemap, convert to UTM zone 10

```
library(ggplot2)
library(PBSmapping)
library(rgdal)

data(nepacLLhigh)
```

Load nepacLLhigh basemap, convert to UTM zone 10

```
library(ggplot2)
library(PBSmapping)
library(rgdal)

data(nepacLLhigh)

# convert basemap to UTM zone 10
attr(nepacLLhigh, "zone") <- 10
str(nepacLLhigh)
```

```
> str(nepacLLhigh) # see projection = "LL", zone = 10
Classes 'PolySet' and 'data.frame': 192762 obs. of 4 variables:
 $ PID: int 0 0 0 0 0 0 0 0 0 ...
 $ POS: int 1 10 17 22 40 53 64 69 70 73 ...
 $ X : num -180 -180 -180 -180 -180 ...
 $ Y : num 69 69 69 69 69 ...
 - attr(*, "PolyData")='data.frame': 0 obs. of 0 variables
 - attr(*, "projection")= chr "LL"
 - attr(*, "zone")= num 10
```

Projection
already specified

UTM zone 10

Load nepacLLhigh basemap, convert to UTM zone 10

```
library(ggplot2)
library(PBSmapping)
library(rgdal)

data(nepacLLhigh)

# convert basemap to UTM zone 10
attr(nepacLLhigh, "zone") <- 10
str(nepacLLhigh)

nepacUTM <- convUL(nepacLLhigh)
```

Simple study area map

```
ggplot() +  
  geom_polygon(data = nepacUTM,  
    aes(x = X, y = Y, group = PID),  
    fill = "grey85", col = "black",  
    lwd = 0.1) +  
  
  coord_equal(xlim = c(430, 540),  
              ylim = c(5330, 5402.5))
```



Simple study area map

Basemap in UTM zone 10

```
ggplot() +  
  geom_polygon(data = nepacUTM,  
               aes(x = X, y = Y, group = PID),  
               fill = "grey85", col = "black"  
               lwd = 0.1) +  
  
  coord_equal(xlim = c(430, 540),  
              ylim = c(5330, 5402.5))
```

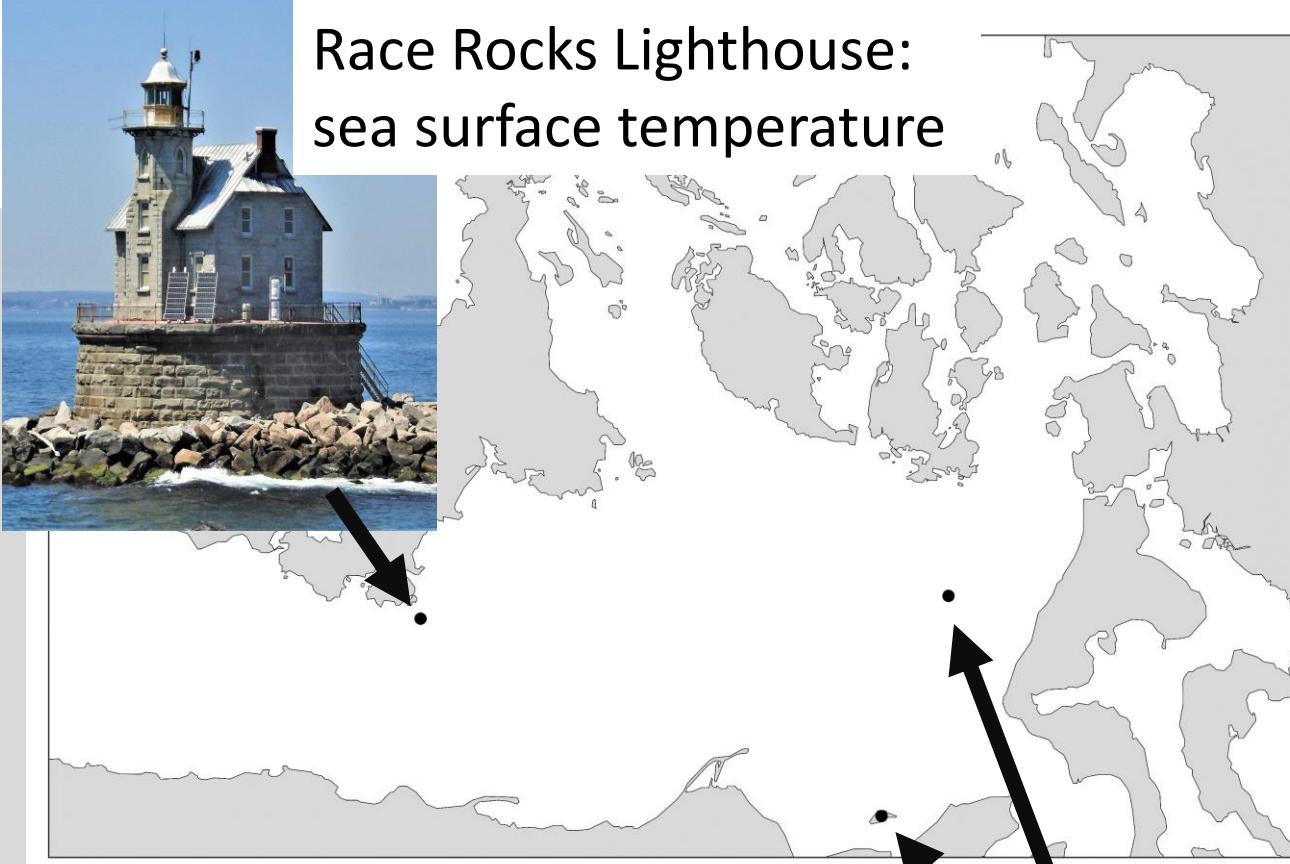
X and Y range of the map



Grouping variable for
polygons – **very important**

Add points of interest

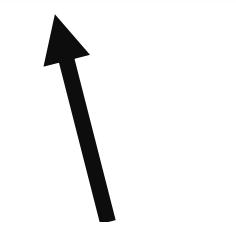
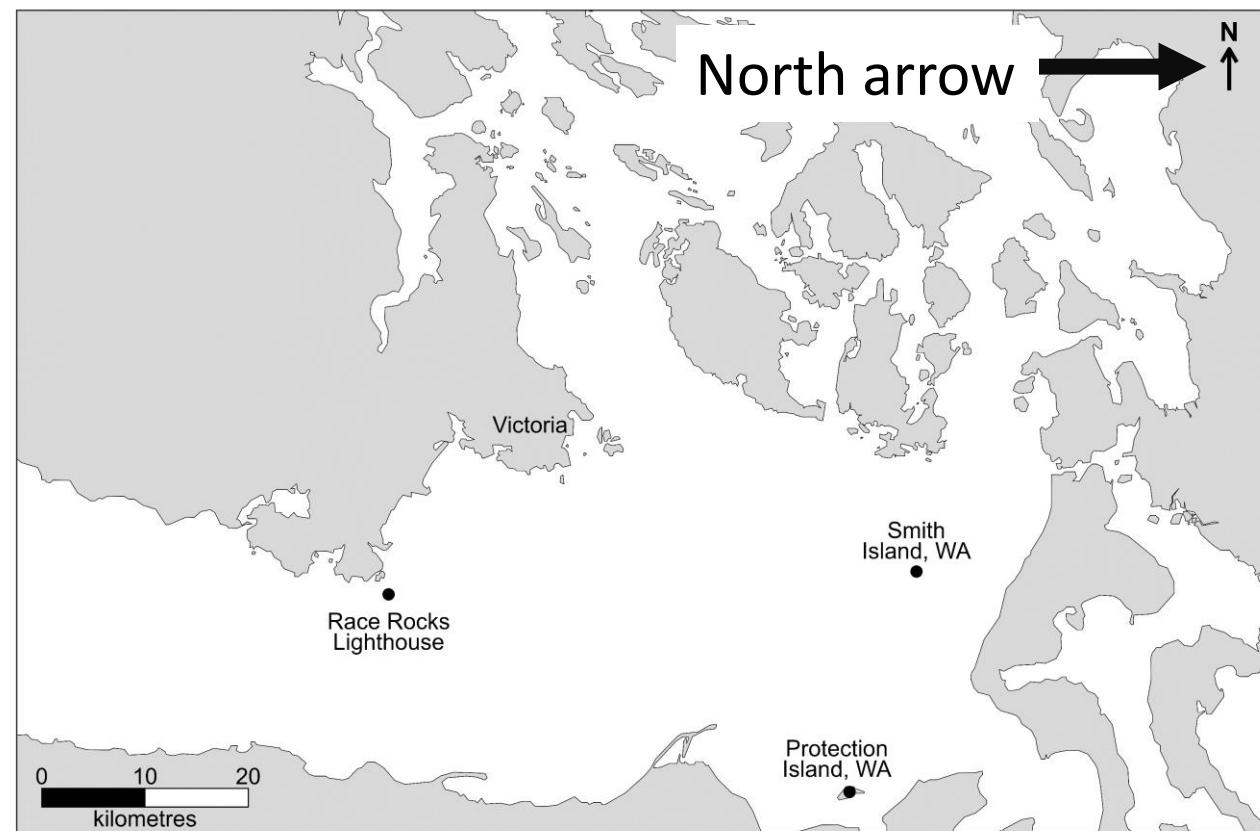
```
ggplot() +  
  geom_polygon(data = nepacUTM,  
    aes(x = X, y = Y, group = PID),  
    fill = "grey85", col = "black",  
    lwd = 0.1) +  
  
  coord_equal(xlim = c(430, 540),  
    ylim = c(5330, 5402.5)) +  
  
  geom_point(data = pointsUTM,  
    aes(x = X, y = Y),  
    col = "black")
```



Important seabird
colonies

Add scalebar, north arrow

```
# add scalebar  
annotate("rect", xmin = 427, y = 447,  
        ymin = 5329, ymax = 5330.8,  
        col = "black", fill = "black",  
        size = 0.2)  
annotate("rect", xmin = 437, y = 447,  
        ymin = 5329, ymax = 5330.8,  
        col = "black", fill = "black",  
        size = 0.2)  
annotate("text", x = 427, y = 5332,  
       label = "0", size = 3) +  
annotate("text", x = 437, y = 5332,  
       label = "10", size = 3) +  
annotate("text", x = 447, y = 5332,  
       label = "20", size = 3) +  
annotate("text", x = 437, y = 5327.9,  
       label = "kilometres", size = 3)
```

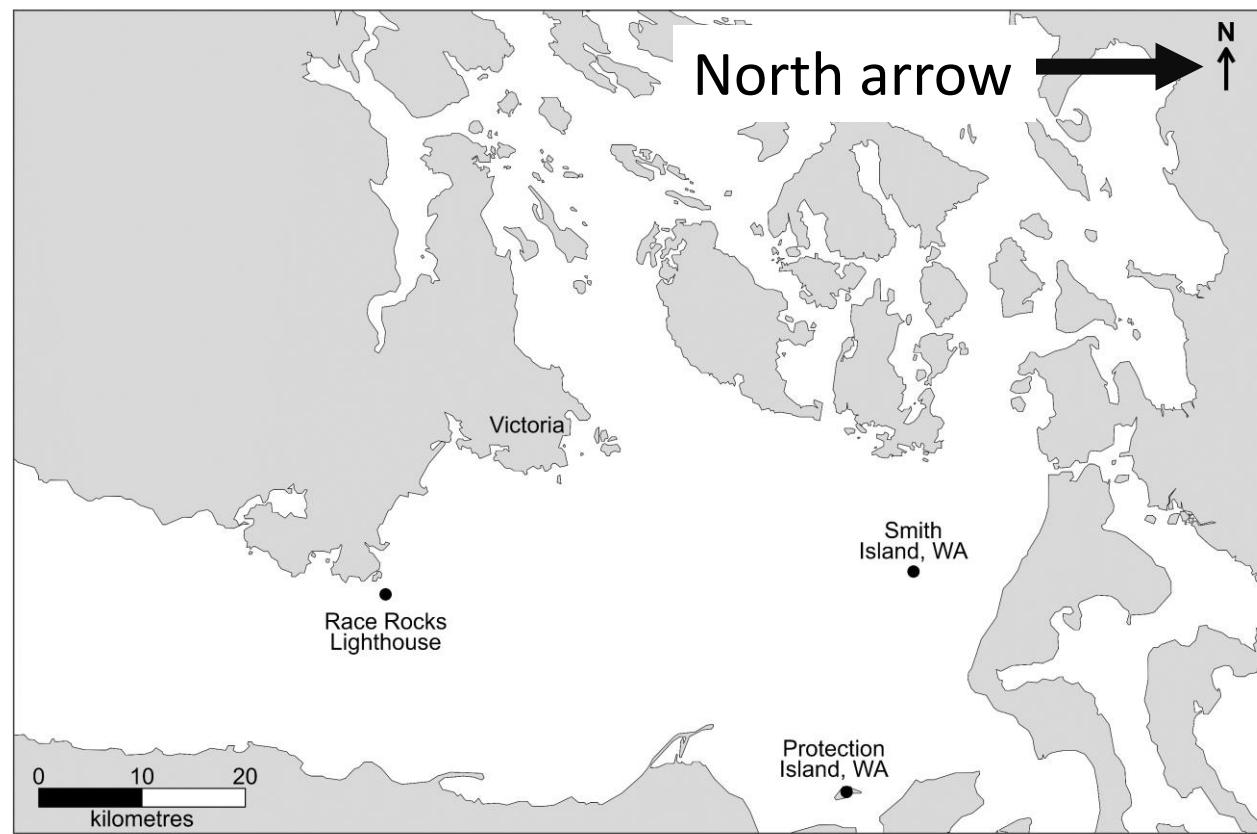


Scalebar

Add scalebar, north arrow

```
# add north arrow  
geom_segment(arrows = arrow(length =  
    unit(0.2, "cm")),  
    aes(x = 542, xend = 542,  
        y = 5398.3, yend = 5402.3),  
    size = 0.7) +  
annotate("text", x = 542, y = 5404,  
    label = "N", fontface = "bold",  
    size= 3.5) +
```

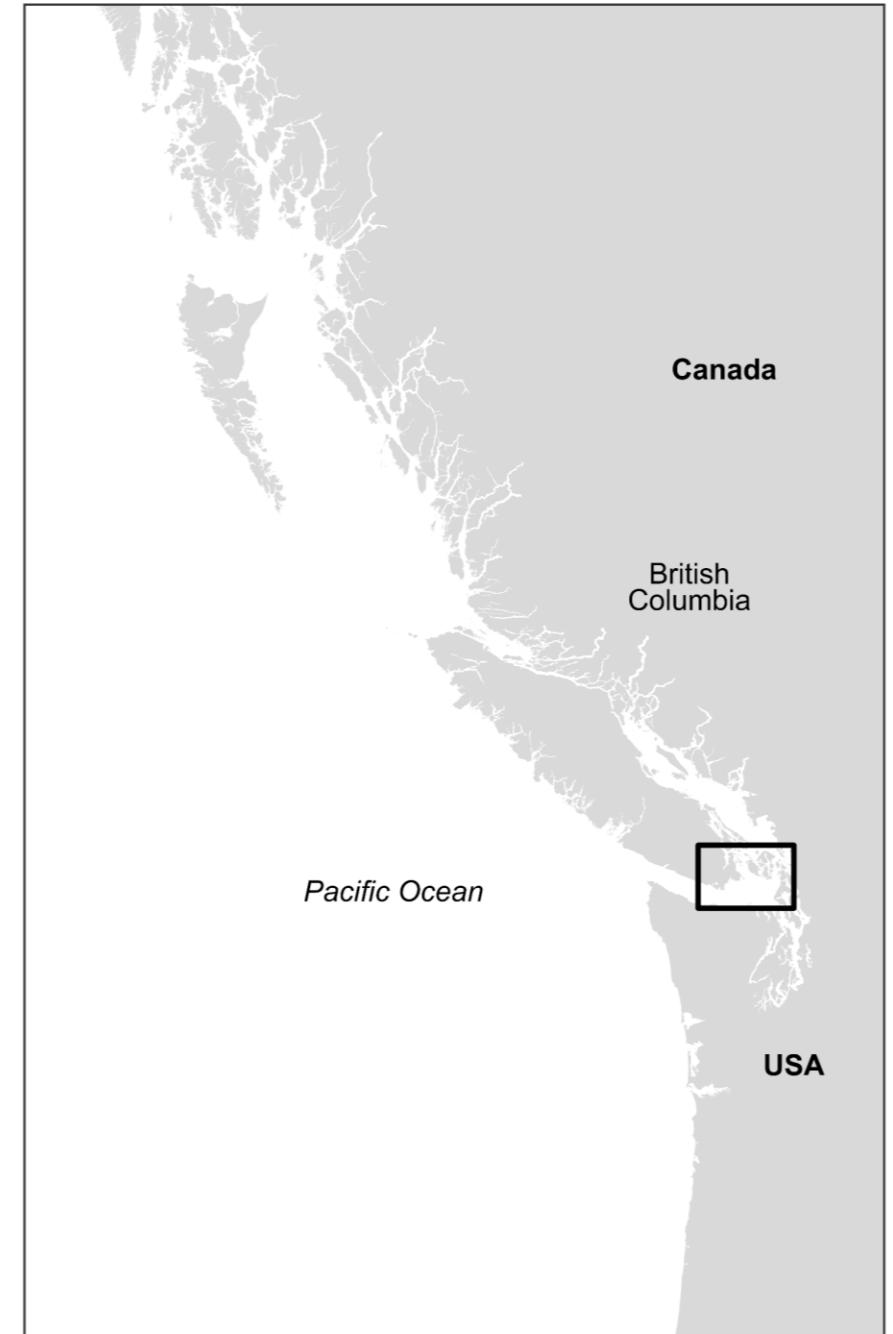
North arrow is straight up for UTM zone 10, but not for some lat/long projections



Scalebar

Add inset map

```
# inset map  
ne.pacific <- ggplot() +  
  geom_polygon(data = nepacUTM,  
    aes(x = X, y = Y, group = PID),  
    fill = "grey85", col = "grey25",  
    lwd = 0.0005) +  
  coord_equal(xlim = c(-300, 600),  
              ylim = c(4900, 6300))
```

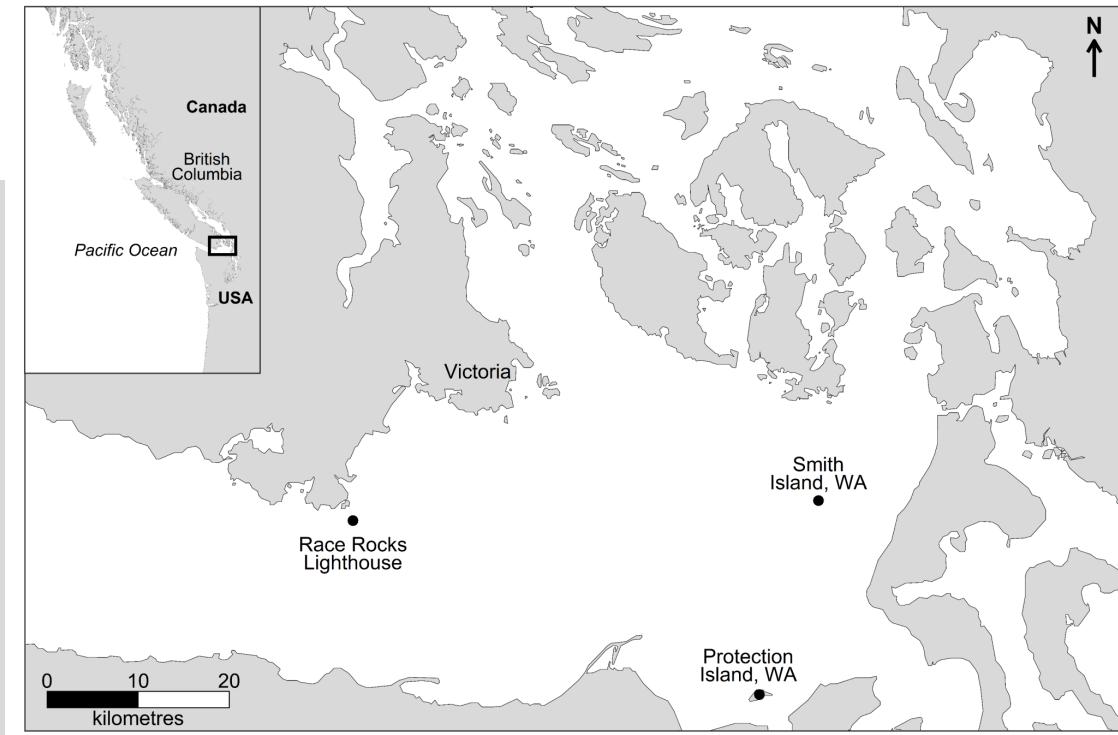


Add inset map

```
# add inset map
tiff("figures/simple-study-area.tiff",
      res = 1600, width = 17, height = 12,
      units = "cm", bg = "white",
      compression = "lzw")
inset.viewport <- viewport(width = 0.5,
                           height = 0.5, x = -0.1316, y = 0.9734,
                           just = c("left", "top"))

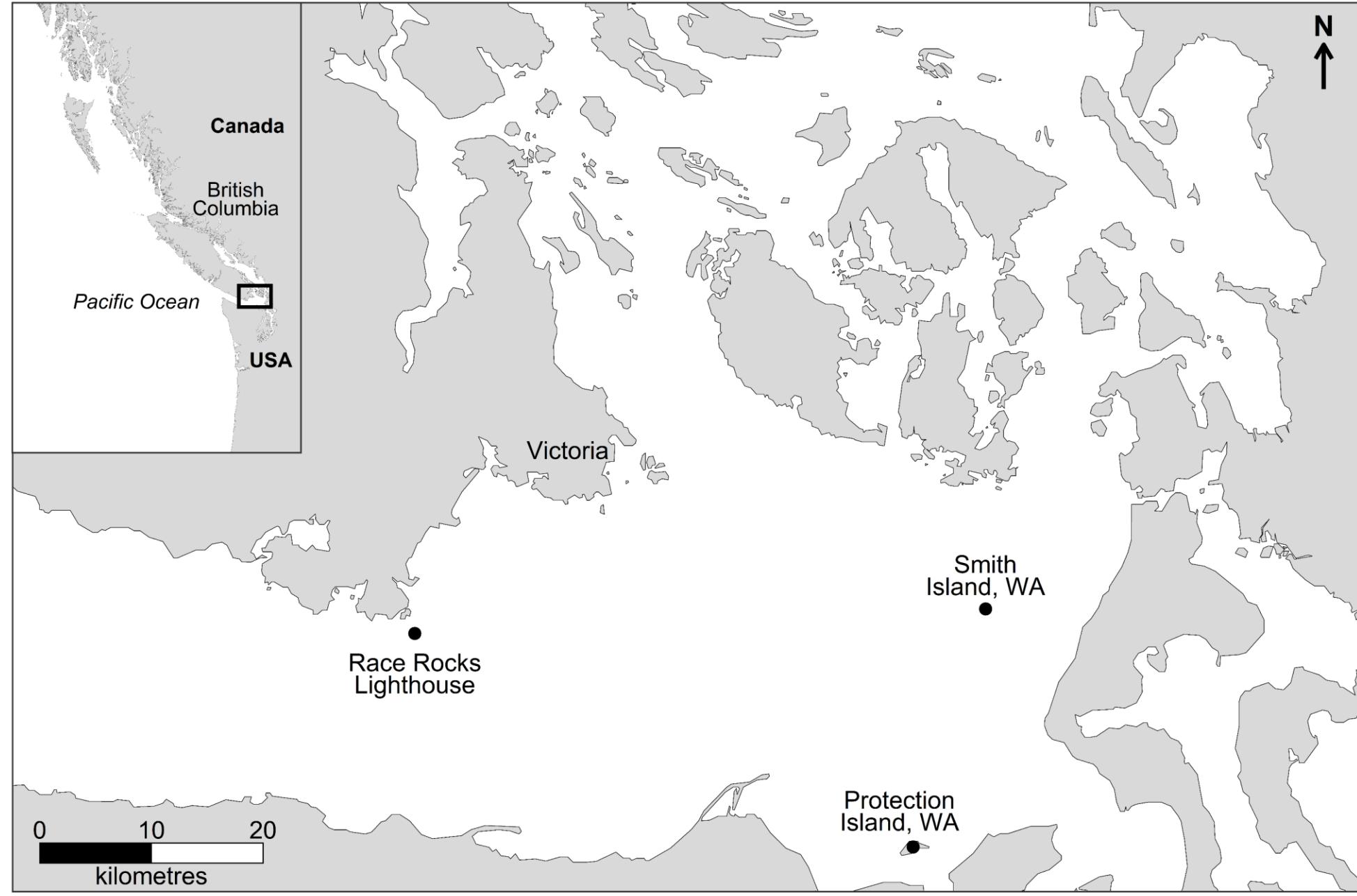
print(haro.strait)
print(ne.pacific, vp = inset.viewport)

dev.off()
```



width, height, x, y
Determine by trial and error so
that inset map is desired size
and location

Add inset map



Where to find additional data layers?

Lots of open-source spatial data available

- BC Government Data Catalogue
<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/topography DEM)
<https://www.gebco.net/>

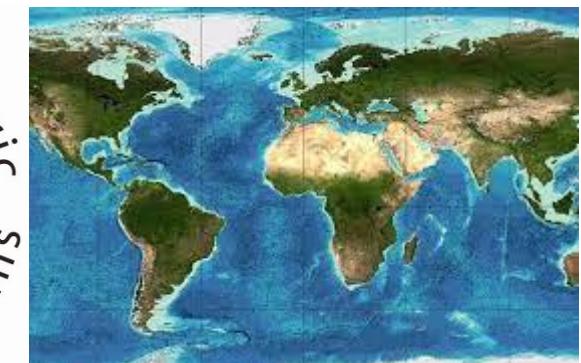


BRITISH
COLUMBIA

DataBC



STRAIT OF GEORGIA
DATA CENTRE



Add rockfish conservation areas

Read in RCA shapefile

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

Note: if sharing shapefiles, need to also include the other files (.cpg, .dbf, .prj, .shx)

```
rca <- rgdal::readOGR("rca-shapefile", "rockfish_102001")
```



Hakai

Add rockfish conservation areas

What projection is the RCA shapefile?



```
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\",  
CRS[\"NAD83\"],\nDATUM[\"North American Datum 1\"] _truncated_  
.. $ comment: chr \"TRUE\"\n> |
```

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\",  
CRS[\"NAD83\"],\n          DATUM[\"North American Datum 1\"] __truncated__  
... $ comment: chr \"TRUE\"\n> |
```

Canada Albers Equal Area Conic

Add rockfish conservation areas



Geoprocessing (this is really hard!)

```
115 # convert to lat/long (WGS 84)
116 rca.wgs <- spTransform(rca, CRS(SRS_string = "EPSG:4326"))
117
118 rca.wgs.coords <- lapply(rca.wgs@polygons,
119                           function(p) data.frame(p@Polygons[[1]]@coords))
120
121 rca.wgs@data$NAME
122
123 # for loop to pull list of polygon geometry into XY format for PBSmapping
124
125 # empty dataframe to add each polygon's XY points into
126 df <- as.data.frame(matrix(nrow = 0, ncol = 3))
127
128 for (i in 1:length(rca.wgs@data$NAME)) {
129   sub <- rca.wgs.coords[[i]] # subset by RCA
130   sub$group <- paste(rca.wgs@data$NAME[i], i, sep = "-")
131   # add i to end because some RCAs with same name
132   df <- rbind(df, sub)
133 }
134
```

This solution is specific to this shapefile

We have to convert the geometry of each polygon into XY format so we can plot with `geom_polygon()`

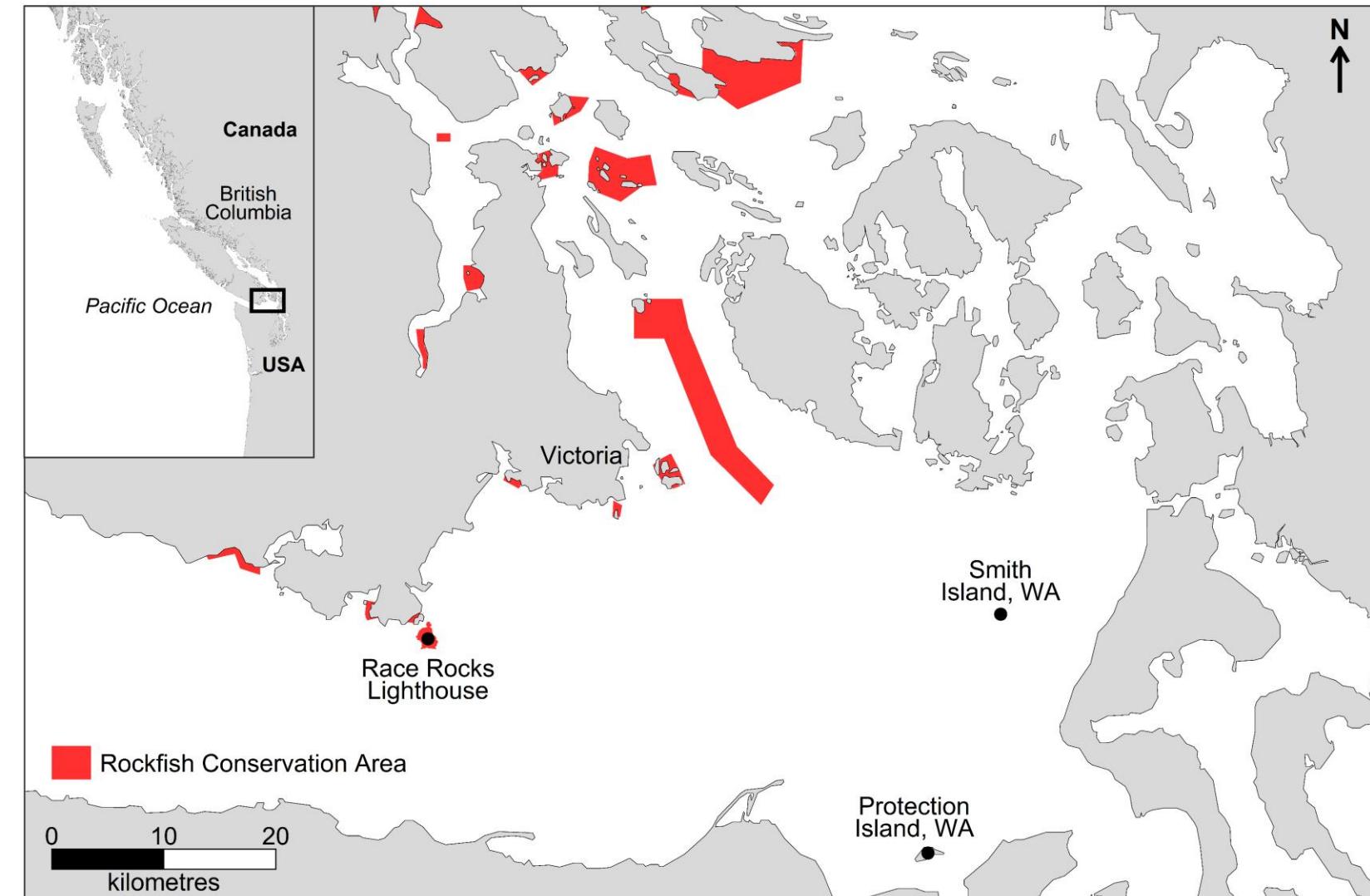
Add rockfish conservation areas

```
geom_polygon(data = rcaUTM, aes(x = X, y = Y,  
group = RCA), fill = "firebrick1") +
```

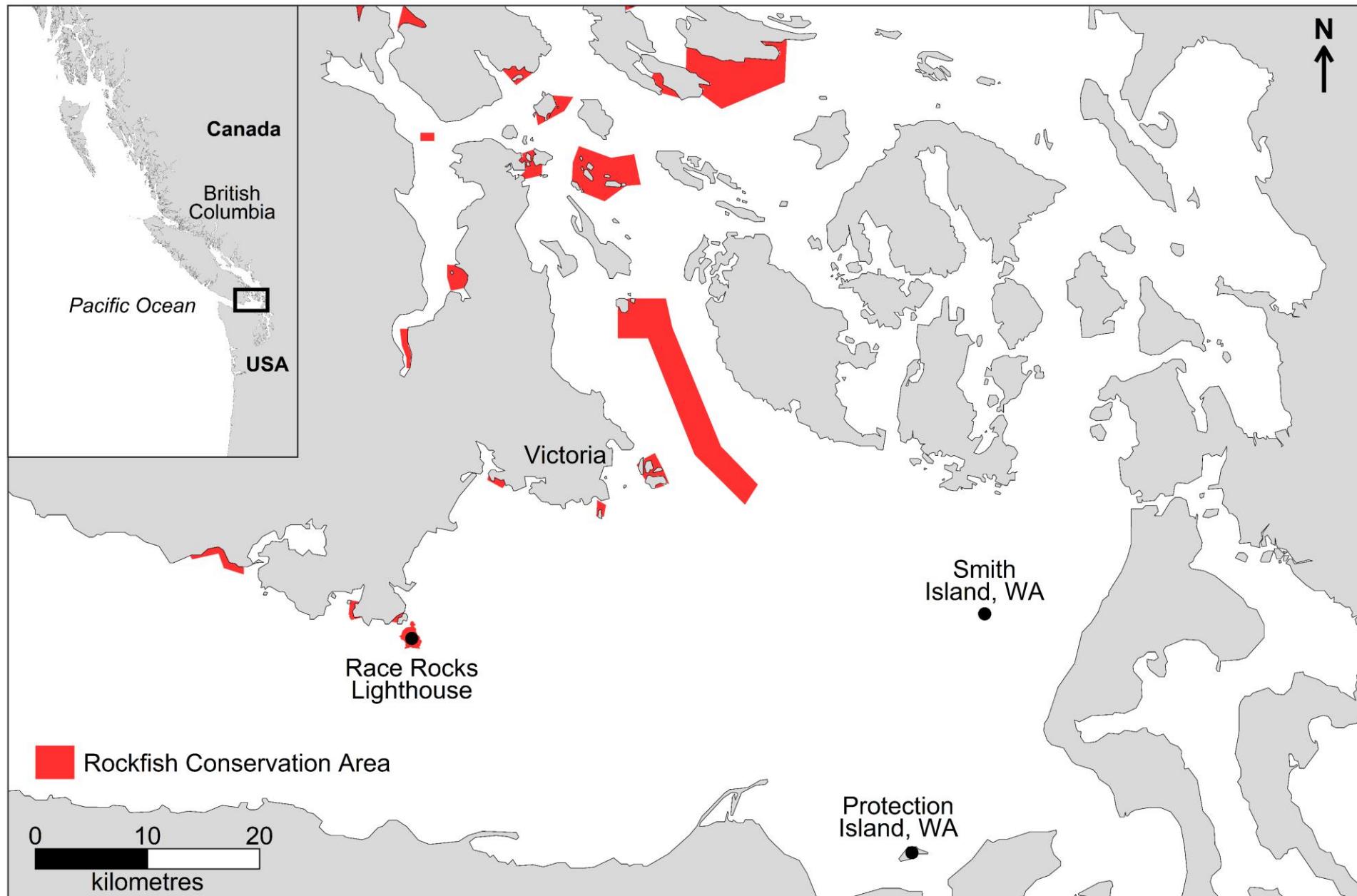


Add rockfish conservation areas

```
geom_polygon(data = rcaUTM, aes(x = X, y = Y,  
group = RCA), fill = "firebrick1") +
```



Add rockfish conservation areas



Save the map

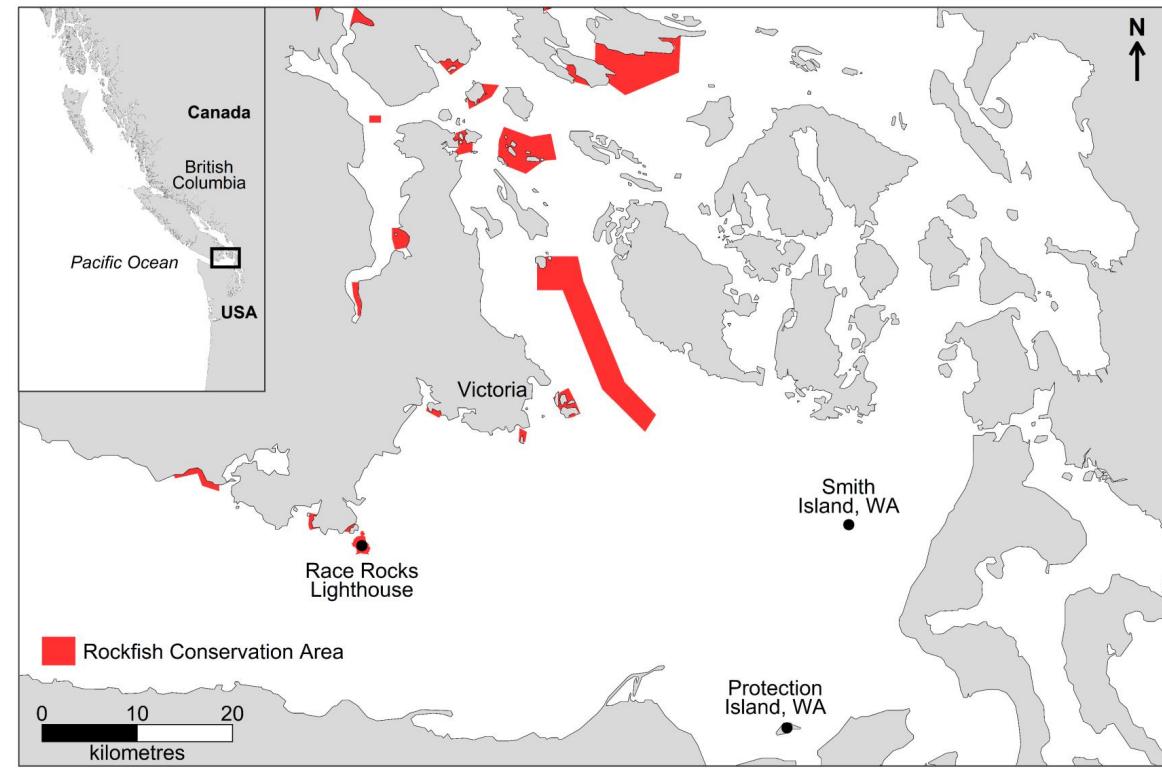
Need to use `tiff()` instead of
`ggsave()` to add inset map

```
tiff("figures/study-area-rca.tiff",
  res = 1600, width = 17, height = 12,
  units = "cm", bg = "white",
  compression = "lzw")
```

```
inset.viewport <- viewport(width = 0.5,
  height = 0.5, x = -0.1316,
  y = 0.9734, just = c("left", "top"))
```

```
print(haro.strait)
print(ne.pacific, vp = inset.viewport)
```

```
dev.off()
```

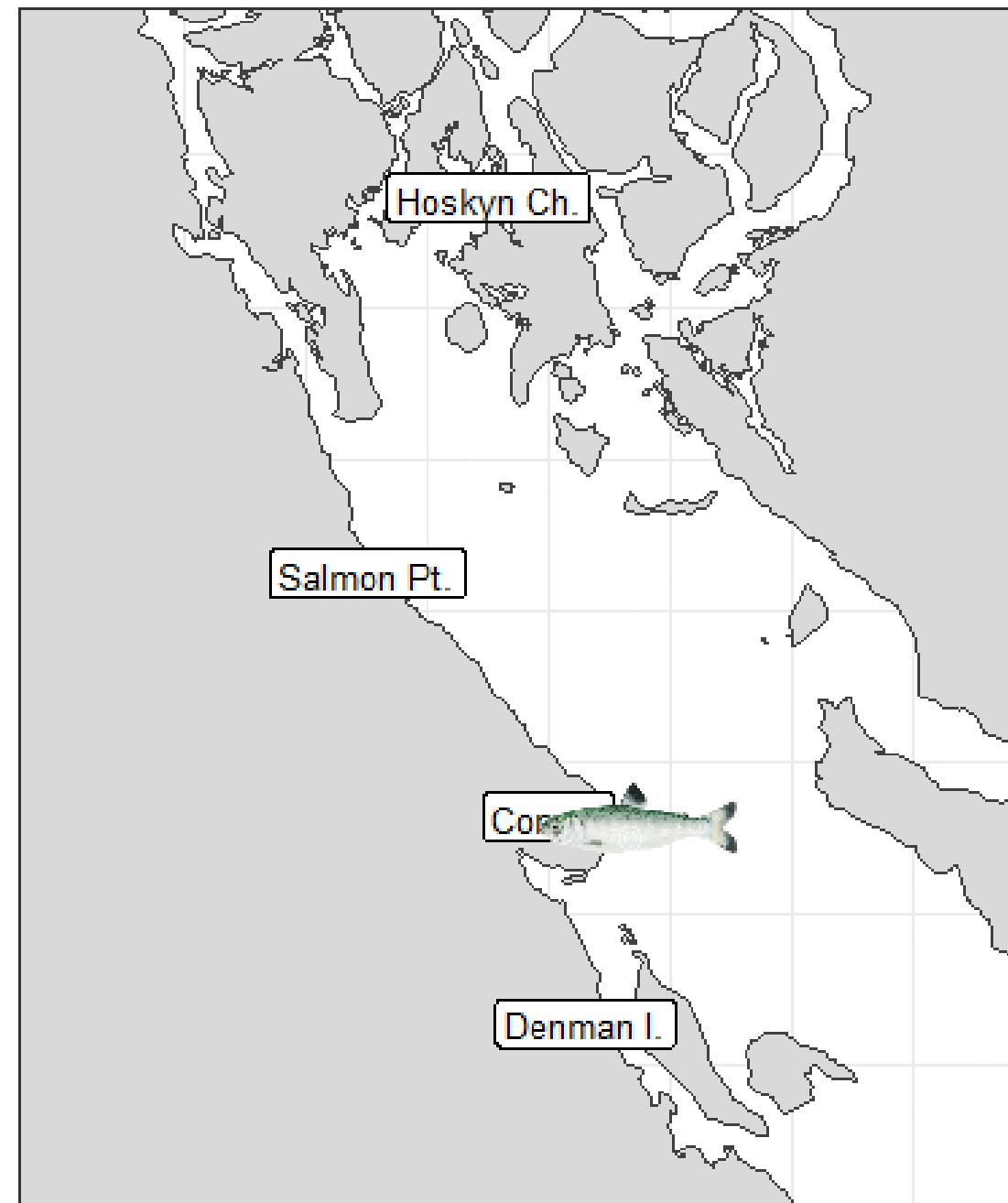


“Advanced topics”

Animated maps using ganimate

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

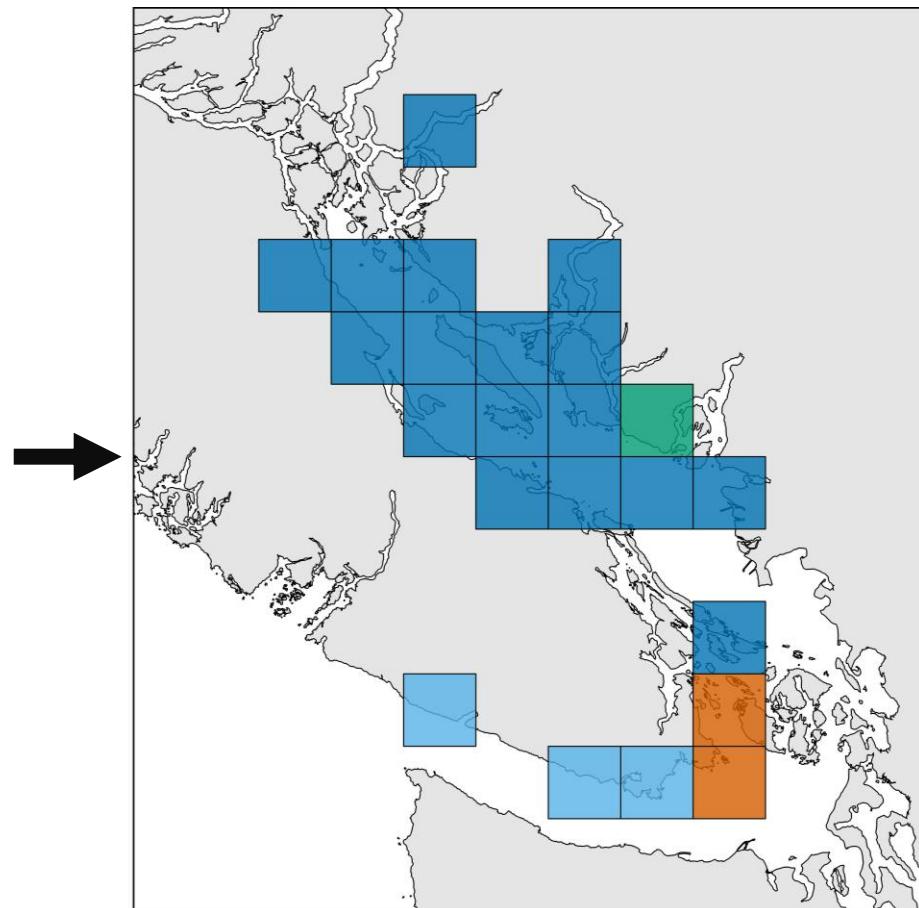
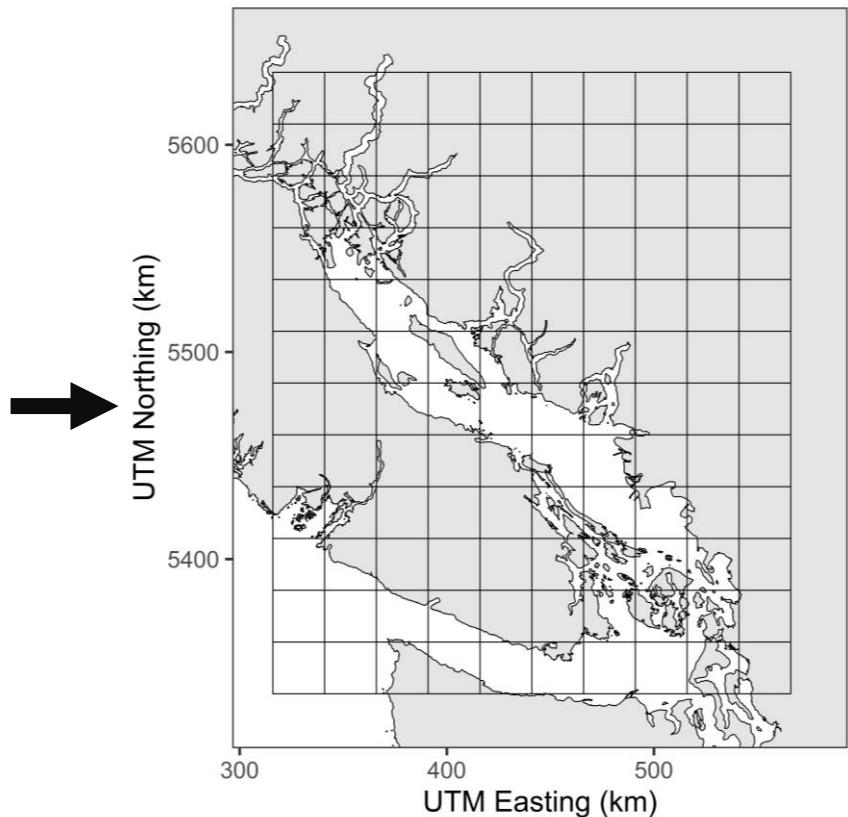
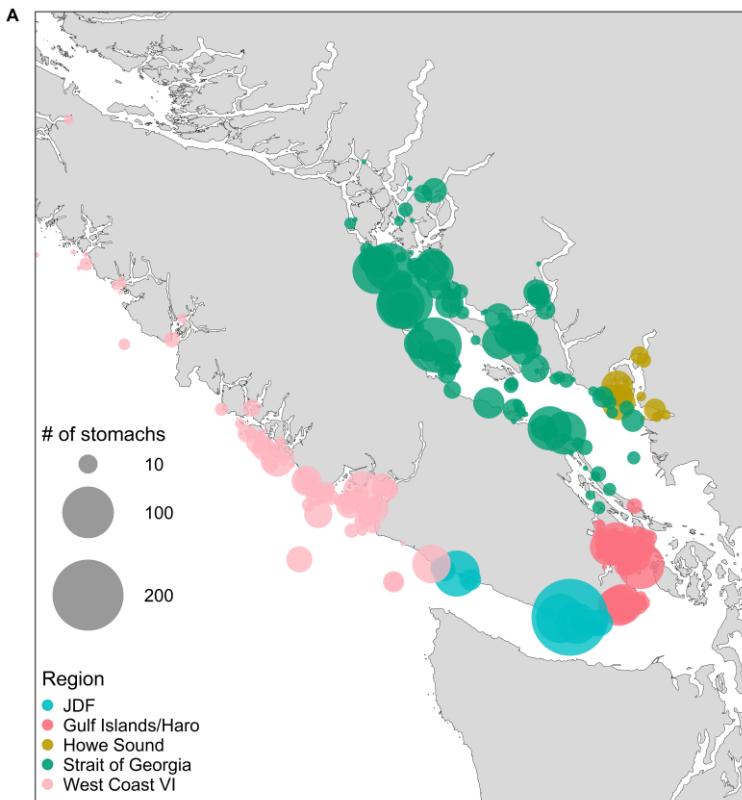
Date/Time: 2023-01-22 11:48:03



“Advanced topics”

Need to use projection in metres, not latitude/longitude

Spatial analysis with grids



`PBSmapping::makeGrid()`

acknowledgements()

PEEC Organizing Committee

Canadian Institute of Ecology and Evolution

Francis Juanes, Will Duguid, and Juanes/Baum labmates



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 PEEC!



Let's make our own maps!

