Ocean Acidification Monitoring Inventory Gap Analysis

## Step 1. Manipulate sea surface temperature and dissolved oxygen as proxies for ocean acidificiation change

#### Load packages

if (!require(pacman)) install.packages("pacman")  
library(pacman)  
p\_load(  
 tidyverse, here, glue,  
 raster,  
 sdmpredictors, dismo,   
 deldir,   
 mapview,  
 tmap)  
  
devtools::load\_all(here("../oatools"))

#### Set paths and variables

dir\_data <- here("data")  
dir\_sdmdata\_old <- here("data/sdmpredictors")  
dir\_cache <- here("cache")  
dir\_sdmdata <- here("cache/sdmpredictors")  
  
SST\_tif <- here("data/sst\_mean.tif")  
DO\_tif <- here("data/do\_mean.tif")

#### Set cache

if (!dir.exists(dir\_data)) dir.create(dir\_data)  
if (!dir.exists(dir\_cache)) dir.create(dir\_cache)  
if (!dir.exists(dir\_sdmdata) & dir.exists(dir\_sdmdata\_old))  
 file.rename(dir\_sdmdata\_old, dir\_sdmdata)  
if (!dir.exists(dir\_sdmdata)) dir.create(dir\_sdmdata)

#### Set extent and coordinate reference system

ext\_study <- extent(-670000, 350000, -885000, 1400000)  
crs\_study <- '+init=EPSG:6414'

#### Create sea surface temperature layer (mean and range)

r\_sst\_mean\_nofill <- lyr\_to\_tif(  
 lyr = "BO\_sstmean",   
 tif = here("data/sst\_mean.tif"),  
 crs = crs\_study,  
 dir\_sdm\_cache = dir\_sdmdata,  
 extent\_crop = ext\_study,   
 redo=T, fill\_na=FALSE)  
  
r\_sst\_mean <- lyr\_to\_tif(  
 lyr = "BO\_sstmean",   
 tif = here("data/sst\_mean.tif"),  
 crs = crs\_study,  
 dir\_sdm\_cache = dir\_sdmdata,  
 extent\_crop = ext\_study,   
 redo=T, fill\_na=TRUE, fill\_window=11) #caclulate mean  
  
n\_na\_nofill <- sum(is.na(raster::getValues(r\_sst\_mean\_nofill)))  
n\_na <- sum(is.na(raster::getValues(r\_sst\_mean)))  
  
r\_sst\_range\_nofill <- lyr\_to\_tif(  
 lyr = "BO\_sstrange",   
 tif = here("data/sst\_range.tif"),  
 crs = crs\_study,  
 dir\_sdm\_cache = dir\_sdmdata,  
 extent\_crop = ext\_study,   
 redo=T, fill\_na=FALSE)  
  
r\_sst\_range <- lyr\_to\_tif(  
 lyr = "BO\_sstrange",   
 tif = here("data/sst\_range.tif"),  
 crs = crs\_study,  
 dir\_sdm\_cache = dir\_sdmdata,  
 extent\_crop = ext\_study,   
 redo=T, fill\_na=TRUE, fill\_window=11)

#### Create dissolved oxygen layer (mean and range)

r\_do\_mean\_nofill <- lyr\_to\_tif(  
 lyr = "BO\_dissox",   
 tif = here("data/do\_mean.tif"),  
 crs = crs\_study,  
 dir\_sdm\_cache = dir\_sdmdata,  
 extent\_crop = ext\_study,   
 redo=T, fill\_na=FALSE)  
  
r\_do\_mean <- lyr\_to\_tif(  
 lyr = "BO\_dissox",   
 tif = here("data/do\_mean.tif"),  
 crs = crs\_study,  
 dir\_sdm\_cache = dir\_sdmdata,  
 extent\_crop = ext\_study,   
 redo=T, fill\_na=TRUE, fill\_window=11) #calculate mean  
  
r\_do\_range\_nofill <- lyr\_to\_tif(  
 lyr = "BO2\_dissoxrange\_bdmin",   
 tif = here("data/do\_range.tif"),  
 crs = crs\_study,  
 dir\_sdm\_cache = dir\_sdmdata,  
 extent\_crop = ext\_study,   
 redo=T, fill\_na=FALSE)  
  
r\_do\_range <- lyr\_to\_tif(  
 lyr = "BO2\_dissoxrange\_bdmin",   
 tif = here("data/do\_range.tif"),  
 crs = crs\_study,  
 dir\_sdm\_cache = dir\_sdmdata,  
 extent\_crop = ext\_study,   
 redo=T, fill\_na=TRUE, fill\_window=11)

## Step 2. Relate SST and DO trends to each monitoring site

#### Load monitoring inventory

oahfocus <- read\_csv(here("data/oahfocus.csv"))

#### Transform into spatial data

coords <- cbind.data.frame(oahfocus$Longitude, oahfocus$Latitude) #isolate coordinate columns  
  
deduped.coords<-unique(coords) # remove duplicate locations  
  
inventorycoords <- SpatialPoints(deduped.coords, CRS("+proj=longlat +ellps=WGS84"))  
inventorycoords <- spTransform(inventorycoords, CRS('+init=EPSG:6414')) # create spatial points objects

#### Create voronoi polygons

vor <-voronoi(inventorycoords)  
  
vorraster<- rasterize(vor, r\_sst\_mean, "id") #rasterize

#### Extract SST mean and range for each monitoring site cell and substitute value for each voronoi polygon

sitesst<- raster::extract(r\_sst\_mean, inventorycoords, method='simple', df=TRUE)  
colnames(sitesst)<-c("id", "SST") #rename column names of sitesst  
polygonsst <- subs(vorraster, sitesst, by="id", which="SST", subsWithNA=FALSE) #substitute polygon id for monitoring site sea surface temerature of that polygon  
  
sitesstrange<- raster::extract(r\_sst\_range, inventorycoords, method='simple', df=TRUE)  
colnames(sitesstrange)<-c("id", "SSTrange") #rename column names of sitesstrange  
polygonsstrange<-subs(vorraster, sitesstrange, by="id", which="SSTrange", subsWithNA=FALSE) #substitute polygon id for monitoring site sea surface temerature range of that polygon

#### Repeat with DO

sitedo<- raster::extract(r\_do\_mean, inventorycoords, method='simple', df=TRUE)  
colnames(sitedo)<-c("id", "DO") #rename column names of sitedo  
polygondo<-subs(vorraster, sitedo, by="id", which="DO")  
  
sitedorange<- raster::extract(r\_do\_range, inventorycoords, method='simple', df=TRUE)  
colnames(sitedorange)<-c("id", "DOrange")  
polygondorange<-subs(vorraster, sitedorange, by="id", which="DOrange")

## Step 3. Create “oceanographic variability” layer relative to monitoring assett

To account for variation in both SST and DO as a combined proxy for ocean acidification variability, we use the following formula: variation = (imean - amean) + (imean - amean)\*(irange - arange) where i = cell in raster of study area and a = cell containing nearest monitoring site

sstmeandiff <- abs(r\_sst\_mean\_nofill - polygonsst) #difference between sst at each point and sst for area associated with closest monitoring site  
sstrangediff <- abs(r\_sst\_range\_nofill - polygonsstrange) #difference between sst range at each point and sst range for area associated with closest monitoring site   
sstvariation <- sstmeandiff+(sstmeandiff\*sstrangediff)  
  
domeandiff <- abs(r\_do\_mean\_nofill - polygondo) #difference between DO at each point and DO for area associated with closest monitoring site  
dorangediff <- abs(r\_do\_range\_nofill - polygondorange) #difference between DO range at each point and DO range for area associated with closest monitoring site  
dovariation <- domeandiff + (domeandiff\*dorangediff)  
  
variation <- (sstvariation\*dovariation) #total variation

## Step 4. Identify gaps

#### Identify gaps based on distance between monitoring points, qualified by strength of variability

distance<-distanceFromPoints(variation,inventorycoords)  
gaps <- setValues(distance, log10(getValues(distance)\*(getValues(variation))))

#### Visualize gaps along the West Coast

poly\_coast<- readOGR(dsn=path.expand("/Users/Madi/Documents/UCSB Bren/ResilienSeas/Export\_Output\_2"), layer="Export\_Output\_2")  
poly\_coast <- spTransform(poly\_coast, crs(gaps))  
gaps\_clipped <- mask(gaps, poly\_coast, inverse = TRUE,progress='text')