KrigingCatalina\_AllBCM

This R Markdown script can be used to interpolate the climate variables for Santa Catalina to the entire extent of the island, as the island is missing BCM climate data for some of its extent. The island DEM was used for universal co-kriging. The model for loops through the climate layers for current climate and all future climate projections and time periods, and saves the interpolated climate layers in a new folder for input into the ArcGIS data processing model. The layers were resampled to 270m, which can be selected in the script.

A big thanks to Allison Horst and her ESM 244 Data Analysis class for teaching us kriging.

#Make sure all packages installed before running- use install.packages("packagename") if necessary  
library(tidyverse) #datascience package

## -- Attaching packages -------------------------------------------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.0.0 v purrr 0.2.5  
## v tibble 1.4.2 v dplyr 0.7.6  
## v tidyr 0.8.1 v stringr 1.3.1  
## v readr 1.1.1 v forcats 0.3.0

## -- Conflicts ----------------------------------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(raster) #raster library

## Loading required package: sp

##   
## Attaching package: 'raster'

## The following object is masked from 'package:dplyr':  
##   
## select

## The following object is masked from 'package:tidyr':  
##   
## extract

library(sf) #spatial

## Linking to GEOS 3.6.1, GDAL 2.2.3, proj.4 4.9.3

library(sp) #more spatial  
library(gstat) #For geospatial statistics

## Warning: package 'gstat' was built under R version 3.5.2

library(rgdal) #spatial library

## rgdal: version: 1.3-4, (SVN revision 766)  
## Geospatial Data Abstraction Library extensions to R successfully loaded  
## Loaded GDAL runtime: GDAL 2.2.3, released 2017/11/20  
## Path to GDAL shared files: C:/Program Files/R/R-3.5.1/library/rgdal/gdal  
## GDAL binary built with GEOS: TRUE   
## Loaded PROJ.4 runtime: Rel. 4.9.3, 15 August 2016, [PJ\_VERSION: 493]  
## Path to PROJ.4 shared files: C:/Program Files/R/R-3.5.1/library/rgdal/proj  
## Linking to sp version: 1.3-1

library(dplyr) #data wrangling   
library(rgeos) #open source

## rgeos version: 0.3-28, (SVN revision 572)  
## GEOS runtime version: 3.6.1-CAPI-1.10.1 r0   
## Linking to sp version: 1.3-1   
## Polygon checking: TRUE

#Define directories and files and pre-processing  
########################################################################  
#Bring in extent data to create spatial grid for Catalina  
dsn\_extent="G:/data/islands/sca/sca\_extent"   
layer\_extent="sca\_extent"  
CI\_extent<-st\_read(dsn = dsn\_extent, layer = layer\_extent)

## Reading layer `sca\_extent' from data source `G:\data\islands\sca\sca\_extent' using driver `ESRI Shapefile'  
## Simple feature collection with 127 features and 2 fields  
## geometry type: POLYGON  
## dimension: XY  
## bbox: xmin: 129579.3 ymin: -522748.8 xmax: 158064.3 ymax: -502967.3  
## epsg (SRID): NA  
## proj4string: +proj=aea +lat\_1=34 +lat\_2=40.5 +lat\_0=0 +lon\_0=-120 +x\_0=0 +y\_0=-4000000 +datum=NAD83 +units=m +no\_defs

#DEM in correct projections  
DEM\_sca<-raster("G:/data/islands/sca/DEM/sca\_dem\_alb/sca\_dem\_alb.tif")   
  
#Make CI extent into spatial points (Outline to raster to points for what to interpolate to)  
ras <- raster()  
extent(ras) <- extent(CI\_extent)  
CI\_ras <- rasterize(CI\_extent, ras, 'ShpAreaSqM')  
proj4string(CI\_ras)<-CRS("+proj=aea +lat\_1=34 +lat\_2=40.5 +lat\_0=0 +lon\_0=-120 +x\_0=0 +y\_0=-4000000 +ellps=GRS80 +datum=NAD83 +units=m +no\_defs")  
  
CI\_ras\_res<-raster("G:/data/islands/sca/sca\_extent/scaext\_alb270.tif")#resample for cell size 270x270  
CI\_resamp<-raster::resample(CI\_ras, CI\_ras\_res)  
  
  
CI\_extent\_points<-rasterToPoints(CI\_resamp, spatial=FALSE)  
CI\_extent\_points<-as.data.frame(CI\_extent\_points)  
  
Grid <- subset(CI\_extent\_points, select=c(x,y))  
colnames(Grid)<-c("Long", "Lat")  
coordinates(Grid) <- ~ Long + Lat  
  
# Make predictions over pixels (not single points)  
gridded(Grid) = TRUE  
proj4string(Grid) <- CRS("+proj=aea +lat\_1=34 +lat\_2=40.5 +lat\_0=0 +lon\_0=-120 +x\_0=0 +y\_0=-4000000 +ellps=GRS80 +datum=NAD83 +units=m +no\_defs")  
  
#Extract from DEM for the GRID  
Grid\_DEM<-raster::extract(DEM\_sca, Grid)  
Grid\_sp<-Grid  
Grid\_sp$DEM<-Grid\_DEM  
  
  
#Time to For Loop through BCM variables  
##########################################################  
  
# dsn\_bcm="G:/data/climate/GISModel\_Input/historic/"  
# files <- list.files(path = dsn\_bcm, pattern = "\\.tif$")#From before f for loop-if only want to run one folder  
  
  
  
folders<-list.dirs(path="G:/data/climate/GISModel\_Input", recursive=FALSE)  
for (f in folders){  
dsn\_bcm<-f  
files <- list.files(path = dsn\_bcm, pattern = "\\.tif$")  
 for (c in files){  
 BCM\_ras <- raster(paste0(dsn\_bcm,"/", c))   
 BCM\_ras\_clip <- mask(BCM\_ras, CI\_extent) #Cut raster to bbox extent of islands  
   
 BCM\_points<-rasterToPoints(BCM\_ras\_clip, spatial=TRUE)#Create points from raster (using defined cell size)  
 BCM\_points<-as.data.frame(BCM\_points)  
 colnames(BCM\_points)<-c("Avg", "lat", "long") #change x to long and y to lat  
 coordinates(BCM\_points) <- ~ lat + long #turns data int spatial points df  
 proj4string(BCM\_points) <- CRS("+proj=aea +lat\_1=34 +lat\_2=40.5 +lat\_0=0 +lon\_0=-120 +x\_0=0 +y\_0=-4000000 +ellps=GRS80 +datum=NAD83 +units=m +no\_defs") #basically doing the same thing that crs did before  
   
 BCM\_DEM<-raster::extract(DEM\_sca, BCM\_points) #Pull DEM points using the raster BCM points  
 BCM\_sp<-BCM\_points  
 BCM\_sp$DEM<-BCM\_DEM   
   
 BCM\_vg <- variogram(Avg ~ DEM, BCM\_sp)#Create a variogram  
   
 vg\_fit <- fit.variogram(BCM\_vg, vgm(c("Exp","Sph","Gau")))#Fit a variogram model  
   
 BCM\_krige <- krige(Avg ~ DEM, BCM\_sp, newdata = Grid\_sp, model = vg\_fit)#Krige  
   
 proj4string(BCM\_krige)<-CRS("+proj=aea +lat\_1=34 +lat\_2=40.5 +lat\_0=0 +lon\_0=-120 +x\_0=0 +y\_0=-4000000 +ellps=GRS80 +datum=NAD83 +units=m +no\_defs")  
   
 writeGDAL(BCM\_krige, fname=paste0("G:/data/climate/GIS\_Input\_SCAInterpolated/", basename(f), "/", c), drivername="GTiff")  
   
 }#end c for loop  
} #end f for loop

## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]

## Warning in fit.variogram(object, x, fit.sills = fit.sills, fit.ranges =  
## fit.ranges, : No convergence after 200 iterations: try different initial  
## values?

## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]

## Warning in fit.variogram(object, x, fit.sills = fit.sills, fit.ranges =  
## fit.ranges, : No convergence after 200 iterations: try different initial  
## values?

## Warning in fit.variogram(object, x, fit.sills = fit.sills, fit.ranges =  
## fit.ranges, : No convergence after 200 iterations: try different initial  
## values?

## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]  
## [using universal kriging]