library(RangeShiftR)  
require(sp)  
require(rgdal)  
require(sfheaders)  
require(raster)  
require(SDMSelect)  
require(tidyverse)

# Introduction

The goal of the analysis is to simulate dispersal of cakile edentula following introductions at the ports of “Sydney”, “Melbourne”, “Brisbane” and “Perth”.

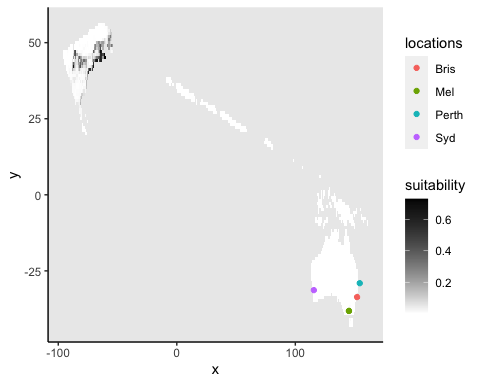
Steps taken:

1. Using the BCCVL platform, and “occurance data” for cakile edentula in its native habitat (in USA) and the climate data “CRUclim (global), current climate (1976-2005), 30 arcmin (~50 km)” as input, predicted a species distribution map for the suitability of cakile edentula. The output is a CSV file with the suitability probabilities (0 - 1) for different points.
2. Transformed the data on suitability from BCCVL into a landscape map file “climate\_suitabilitya.asc”.

#### suitability map  
  
AllData <- read.csv("Inputs/raw SDM.csv")  
colnames(AllData)[1:2] = c("lon", "lat")  
  
coordinates(AllData) = ~ lon + lat  
proj4string(AllData)=CRS("+init=epsg:4326") # set it to lat-long  
AllData = spTransform(AllData,CRS("+init=epsg:4326"))  
  
e <- extent(AllData)  
r <- raster(e, ncols = 1000, nrows = 1000)  
  
# you need to provide a function 'fun' for when there are multiple points per cell  
AllData\_a <- rasterize(AllData, r, fun = mean)  
  
AllData2a = raster::resample(AllData\_a[[2]], raster(ext=extent(extent(AllData\_a)), resolution=1,crs=projection(AllData\_a)))  
  
  
raster::writeRaster(AllData2a, format="ascii", filename = "Inputs/climate\_suitabilitya", NAflag = -9, overwrite = T, bylayer = T, datatype = "FLT4S")  
  
lp = SDMSelect::gplot\_data(raster("Inputs/climate\_suitabilitya.asc")) %>% filter(!is.na(value))

1. Identified the map locations for the ports of “Sydney”, “Melbourne”, “Brisbane” and “Perth” where cakile edentula are introduced. Transformed this data into a species distribution map file “news\_locations.asc”

introduction = read.csv("Inputs/aus\_ports3.csv")[, 1:3]  
introduction$Individuals = as.numeric(introduction$Individuals)  
  
a3 = introduction %>% filter(!is.na(Individuals)) %>% mutate("locations" = c("Syd", "Mel", "Mel", "Bris", "Perth")) %>% na.omit()  
  
coordinates(introduction) = ~ lon + lat  
proj4string(introduction)=proj4string(AllData) # set it to lat-long  
introduction = spTransform(introduction,proj4string(AllData))  
  
e <- extent(AllData\_a)  
r <- raster(e, ncols = 720, nrows = 360)  
  
# you need to provide a function 'fun' for when there are multiple points per cell  
introduction2 <- rasterize(introduction, r, fun = mean)  
  
introduction3 = raster::resample(introduction2[[2]], raster(ext=extent(extent(introduction2)), resolution=2,crs=projection(introduction)))  
  
  
raster::writeRaster(introduction3, format="ascii", filename = "Inputs/news\_locations.asc", NAflag = -9, overwrite = T, datatype = "INT2U")  
  
ggplot( ) +   
 geom\_tile(data = lp , aes(x = x, y = y, fill = value)) +   
 scale\_fill\_gradient("suitability", low = "white", high = "black") +  
 theme\_grey() + theme(panel.border = element\_blank(), panel.grid.major = element\_blank(),  
panel.grid.minor = element\_blank(), axis.line = element\_line(colour = "black")) +   
 geom\_point(data = a3, aes(x = lon, y = lat, color = locations))



*The chart above displays the climate suitability for cakile edentula in the selected region that overlaps between the native area in the USA and Australia. From the map, we observe that suitability is only relatively high in the native region and very low everywhere else. The introduction points for this simulation are also shown and we note that they are all regions with very low suitability.*

1. Define the *landscape parameter* in RangeshiftR for simulation. We define the resolution at for the landscape and 2km for the species distribution maps. We set *HabPercent = TRUE* since the data in our suitability map is a probability value between 0 and 1. We set the *K\_or\_DensDep = 1000* to infer that a location with a probability of 1 would have a carrying capacity of 1000. The carrying capacities for all locations are calculated based on the probability value and the 1000 threshold.

landscapes = ImportedLandscape(LandscapeFile = "climate\_suitabilitya.asc",   
 Resolution = 1,   
 HabPercent = TRUE,   
 K\_or\_DensDep = 1000,   
 SpDistFile = "news\_locations.asc",   
 SpDistResolution = 2)

1. We define the Species parameters which include:

* *demography*: I have made the assumption that the population dynamics are described by a female-only model with non-overlapping generations. I set:
  + the *maximum growth rate (Rmax) = 5*
  + *under-compensatory density regulation dynamics (BC) =* 0.00000001

demos <- Demography(Rmax = 5, bc = 0.00000001, ReproductionType = 0)  
demos

## Demography:  
## Unstructured population:  
## Rmax : 5   
## bc : 1e-08   
## Reproduction Type : 0 (female only)

* *dispersal*: This contains information on *Emigration* probability (0.5), *Transport* modeled with a dispersal kernel (double kernel) and *Settlement* that specifies that if an individual finds a cell it emigrates to unsuitable, it will move to one of the eight neighbouring cells in the case that at least one of them is suitable (Settle= *2*).

We make the assumption that the *Emigration probability = 0.5*

We define *short range* and *long range transfer* at with and with respectively (You provided this values). This suggests that we have a 3rd possible dispersal distance equal to which has a probability of . However, rangeshiftR provides for either a single or double kernel to define distance. To fit this, I considered the odds between the short dispersal distance (20km) and the long distance (100km) and used this to recalculate their probabilities assuming the 3rd option of is not present.

dist = matrix(c(20000, 100000, 0.83 ), ncol = 3, byrow = T)  
  
disps <- Dispersal(Emigration = Emigration(EmigProb = 0.5), Transfer = DispersalKernel(DoubleKernel = T, Distances = dist)) + Settlement(MaxSteps = 5, Settle = 2)  
disps

## Dispersal:   
## Emigration:  
## Emigration probabilities:  
## [,1]  
## [1,] 0.5  
##   
## Transfer:  
## Dispersal Kernel  
## DoubleKernel = TRUE   
## Dispersal kernel traits:  
## [,1] [,2] [,3]  
## [1,] 20000 1e+05 0.83  
## Constant mortality prob = 0   
##   
## Settlement:  
## Settlement conditions:  
## [,1]  
## [1,] 2  
## FindMate = FALSE

1. Initialize the parameters for simulation

inits <- Initialise(InitType = 1, # = initialisation from a loaded species distribution map  
 SpType = 1,# = all suitable cells within all distribution presence cells  
 NrCells = 20,  
 InitDens = 2,  
 IndsHaCell = 100) # = at carrying capacity  
  
inits

## Initialisation:   
## InitType = 1 : Initialisation from loaded species distribution map  
## all presence cells/patches.  
## InitDens = 2 : 100 individuals per cell/hectare

1. Simulation parameters

We will run with simulating the spread of cakile edentula over and record the *population*, *occurance* and *range* every

sims <- Simulation(Simulation = 1,  
 Years = 1000,  
 Replicates = 10,  
 OutIntRange = 5,  
 OutIntPop = 5,  
 OutIntOcc = 5)  
sims

## Simulation # 2   
## -----------------  
## Replicates = 10   
## Years = 1000   
## Absorbing = FALSE   
## File Outputs:  
## Range, every 5 years  
## Occupancy, every 5 years  
## Populations, every 5 years, starting year 0

1. Combine all the predefined parameters into a master object and run the simulation. We will run two separate simulations, one for the short range dispersal and the other for the long range dispersal.

simulate <- RSsim() + landscapes + demos + disps + sims + inits  
validateRSparams(simulate) ## check parameter validity

## [1] TRUE

1. Run simulations

Dispersal of cakile edentula varies between different replicates. From the results I have obtained, simulating dispersal over 1000 years;

* some replicates indicates that starting with 600 individuals at year 0 results in an increase of individuals in 1000 years.
* Some replicates still show that the species dies out but after quite a long time.

set.seed(1000)  
RunRS(RSparams = simulate, dirpath = paste0(getwd(), "/"))

## Checking Control parameters   
##   
## Control Parameters checked  
##   
## Run Simulation(s) with random seed ...  
## LandscapeFile headers OK: /Users/kwizera.jvk/Desktop/RangeShiftR/Simulation\_cakile\_Australia/2. Climate suitability Simulation/results/Inputs/climate\_suitabilitya.asc  
## Species Distribution map headers OK: /Users/kwizera.jvk/Desktop/RangeShiftR/Simulation\_cakile\_Australia/2. Climate suitability Simulation/results/Inputs/news\_locations.asc  
##   
## Running simulation nr. 2  
##   
## starting replicate 0  
## RunModel(): completed initialisation   
## starting year 0...  
## starting year 1...  
## starting year 2...  
## starting year 3...  
## starting year 10...  
## starting year 20...  
## starting year 30...  
## starting year 100...  
## starting year 200...  
## starting year 300...  
##   
## starting replicate 1  
## RunModel(): completed initialisation   
## starting year 0...  
## starting year 1...  
## starting year 2...  
## starting year 3...  
## starting year 10...  
## starting year 20...  
## starting year 30...  
## starting year 100...  
## starting year 200...  
## starting year 300...  
##   
## starting replicate 2  
## RunModel(): completed initialisation   
## starting year 0...  
## starting year 1...  
## starting year 2...  
## starting year 3...  
## starting year 10...  
## starting year 20...  
## starting year 30...  
## starting year 100...  
## starting year 200...  
## starting year 300...  
##   
## starting replicate 3  
## RunModel(): completed initialisation   
## starting year 0...  
## starting year 1...  
## starting year 2...  
## starting year 3...  
## starting year 10...  
## starting year 20...  
## starting year 30...  
## starting year 100...  
## starting year 200...  
## starting year 300...  
##   
## starting replicate 4  
## RunModel(): completed initialisation   
## starting year 0...  
## starting year 1...  
## starting year 2...  
## starting year 3...  
## starting year 10...  
## starting year 20...  
## starting year 30...  
## starting year 100...  
## starting year 200...  
## starting year 300...  
##   
## starting replicate 5  
## RunModel(): completed initialisation   
## starting year 0...  
## starting year 1...  
## starting year 2...  
## starting year 3...  
## starting year 10...  
## starting year 20...  
## starting year 30...  
## starting year 100...  
## starting year 200...  
## starting year 300...  
##   
## starting replicate 6  
## RunModel(): completed initialisation   
## starting year 0...  
## starting year 1...  
## starting year 2...  
## starting year 3...  
## starting year 10...  
## starting year 20...  
## starting year 30...  
## starting year 100...  
## starting year 200...  
## starting year 300...  
##   
## starting replicate 7  
## RunModel(): completed initialisation   
## starting year 0...  
## starting year 1...  
## starting year 2...  
## starting year 3...  
## starting year 10...  
## starting year 20...  
## starting year 30...  
## starting year 100...  
## starting year 200...  
## starting year 300...  
##   
## starting replicate 8  
## RunModel(): completed initialisation   
## starting year 0...  
## starting year 1...  
## starting year 2...  
## starting year 3...  
## starting year 10...  
## starting year 20...  
## starting year 30...  
## starting year 100...  
## starting year 200...  
## starting year 300...  
##   
## starting replicate 9  
## RunModel(): completed initialisation   
## starting year 0...  
## starting year 1...  
## starting year 2...  
## starting year 3...  
## starting year 10...  
## starting year 20...  
## starting year 30...  
## starting year 100...  
## starting year 200...  
## starting year 300...  
##   
## \*\*\*\*\* Elapsed time: 157 seconds  
##   
## \*\*\*\*\*  
## \*\*\*\*\* Simulation completed   
## \*\*\*\*\* Outputs folder: /Users/kwizera.jvk/Desktop/RangeShiftR/Simulation\_cakile\_Australia/2. Climate suitability Simulation/results/Outputs/  
## \*\*\*\*\*

## $Errors  
## [1] 0