

Additional File 8 - calculating distances between capture and release sites

Langridge et al.

09-02-2021

Contents

1. Packages and versions	1
2. Full data	2
2.1. Data management	2
2.2 Mercator projection	2
3. Calculate distances - Pythagore	4
3.1. Calculate distance groups	4
4. Export data table	4

1. Packages and versions

R version 3.3.2 (2016-10-31) – “Sincere Pumpkin Patch” Copyright (C) 2016 The R Foundation for Statistical Computing Platform: x86_64-w64-mingw32/x64 (64-bit)

R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.

```
library(sp)
library(bioge)
library(rgdal)
```

2. Full data

2.1. Data management

```
setwd("C:\\Users\\adrie\\Documents\\NaturAdapt")
my.coords.data <- read.csv2("Calculating Distances.csv", h=T)
```

```
my.coords.data <- my.coords.data[which(my.coords.data$enrNA == 0),]
```

In order to take out any NAs (i.e. missing data) in ones dataset.

```
## capture sites coordinates
my.coords.data$ddcapture_lat_coord <- dms2dd (my.coords.data$dd_cpty,
                                              my.coords.data$mm_cpty,
                                              my.coords.data$ss_cpty,
                                              my.coords.data$h_cpty)

my.coords.data$ddcapture_long_coord<-dms2dd(my.coords.data$dd_cptx,
                                             my.coords.data$mm_cptx,
                                             my.coords.data$ss_cptx,
                                             my.coords.data$h_cptx)

## capture releases coordinates
my.coords.data$ddrelease_lat_coord<-dms2dd(my.coords.data$dd_rpty,
                                             my.coords.data$mm_rpty,
                                             my.coords.data$ss_rpty,
                                             my.coords.data$h_rpty)

my.coords.data$ddrelease_long_coord<-dms2dd(my.coords.data$dd_rptx,
                                             my.coords.data$mm_rptx,
                                             my.coords.data$ss_rptx,
                                             my.coords.data$h_rptx)
```

From the 'biogeo' library - transform DMS to degrees.

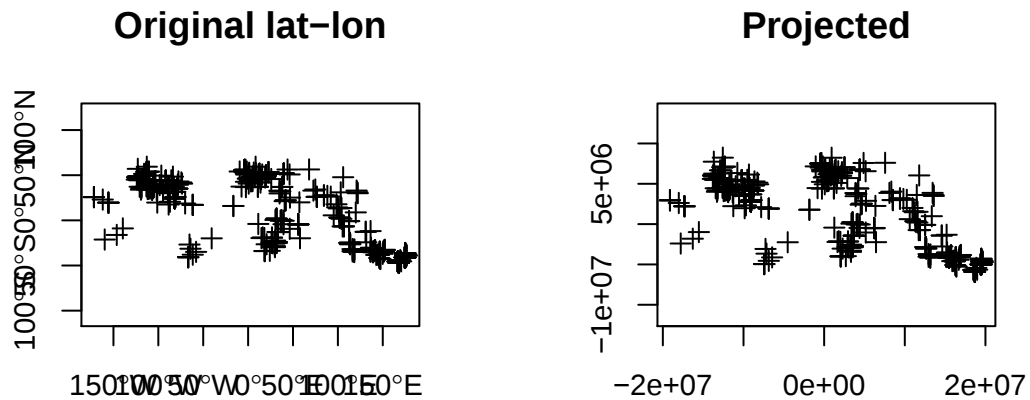
2.2 Mercator projection

```
## Convert capture sites coordinates to meters ##
capture_latlong_coord<-data.frame(ENR=my.coords.data$ENR,
                                   x=my.coords.data$ddcapture_long_coord,
                                   y=my.coords.data$ddcapture_lat_coord)

coordinates(capture_latlong_coord) <- c("x", "y")
proj4string(capture_latlong_coord) <- CRS("+init=epsg:4326") ## WGS84 is the reference

CRS.Merc<- CRS(
  "+proj=merc +lon_0=0 +k=1 +x_0=0 +y_0=0 +ellps=WGS84 +datum=WGS84 +units=m +no_defs")

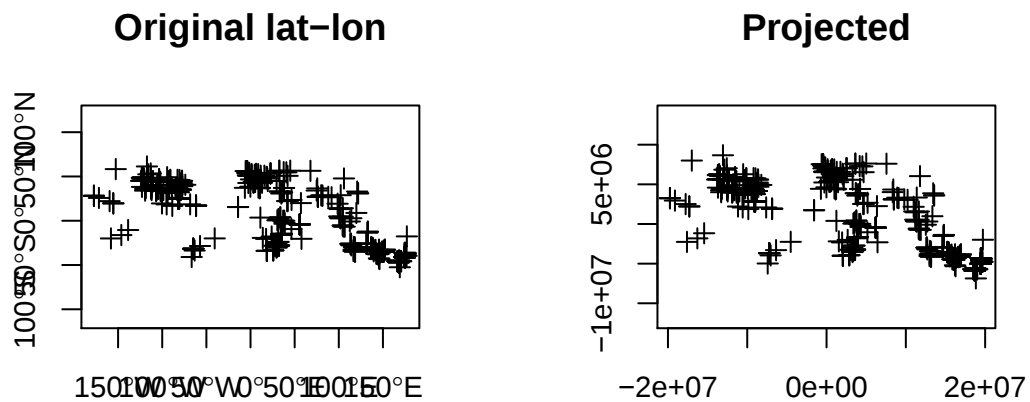
newcoor tot_Merc_c <- spTransform(capture_latlong_coord, CRS.Merc)
par(mfrow=c(1,2))
plot(capture_latlong_coord, axes=TRUE, main="Original lat-lon", cex.axis=.95)
plot(newcoor tot_Merc_c, axes=TRUE, main="Projected", cex.axis=.95)
```



```
my.coords.data$Merc_x_capt_m<-newcoor dtot_Merc_c$x
my.coords.data$Merc_y_capt_m<-newcoor dtot_Merc_c$y

## Convert release sites coordinates to meters ##
release_latlong_coord<-data.frame(enr=my.coords.data$ENR,
                                   x=my.coords.data$ddrelease_long_coord,
                                   y=my.coords.data$ddrelease_lat_coord)
coordinates(release_latlong_coord) <- c("x", "y")
proj4string(release_latlong_coord) <- CRS("+init=epsg:4326") ## WGS84 is the reference

newcoor dtot_Merc_r <- spTransform(release_latlong_coord, CRS.Merc)
par(mfrow=c(1,2))
plot(release_latlong_coord, axes=TRUE, main="Original lat-lon", cex.axis=.95)
plot(newcoor dtot_Merc_r, axes=TRUE, main="Projected", cex.axis=.95)
```



```
my.coords.data$Merc_x_relea_m<-newcoor dtot_Merc_r$x
my.coords.data$Merc_y_relea_m<-newcoor dtot_Merc_r$y
```

3. Calculate distances - Pythagore

```
## with Mercator projection

my.coords.data$Mercator_distances_km <- round(
  sqrt((my.coords.data$Merc_x_relea_m-my.coords.data$Merc_x_capt_m)^2+
        (my.coords.data$Merc_y_relea_m-my.coords.data$Merc_y_capt_m)^2)/1000, 1)
```

3.1. Calculate distance groups

```
my.coords.data$Mercator_grouped <- cut(my.coords.data$Mercator_distances_km,
                                       breaks=c(0,100,500,1000,1500,2000, 2500, 3000,
                                                3500, 4000, Inf),
                                       include.lowest=TRUE,
                                       labels=c("0-100", "101-500", "501-1000", "1001-1500",
                                                "1501-2000", "2001-2500", "2501-3000", "3001-3500",
                                                "3501-4000", ">4000"))

View (my.coords.data)
```

4. Export data table

```
write.table(my.coords.data, "Calculated_distances_VF.csv", sep=",")
```

Acknowledgements: We would like to sincerely acknowledge the assistance and expertise of Dr. Benoit PISANU of the Natural heritage service UMS PatriNat (OFB-CNRS-MNHN)