test isOrigin.Oct.9.2018

Chao Ma

Oct. 9, 2018

Install isOrigin package from Github

```
devtools::install_github("SPATIAL-Lab/isorig", force=T)
```

```
## Downloading GitHub repo SPATIAL-Lab/isorig@master
## from URL https://api.github.com/repos/SPATIAL-Lab/isorig/zipball/master
```

```
## Installing isOrigin
```

```
## "C:/PROGRA~1/R/R-35~1.1/bin/x64/R" --no-site-file --no-environ --no-save
\
##   --no-restore --quiet CMD INSTALL "C:/Users/Chao  \
##   Ma/AppData/Local/Temp/RtmpGQ5Mij/devtoolsb6020554423/SPATIAL-Lab-isorig-
90c074f"  \
##   --library="C:/Users/Chao Ma/Documents/R/win-library/3.5"  \
##   --install-tests

##
```

Load library

```
library(isOrigin)
```

```
## Loading required package: raster
```

```
## Loading required package: sp
```

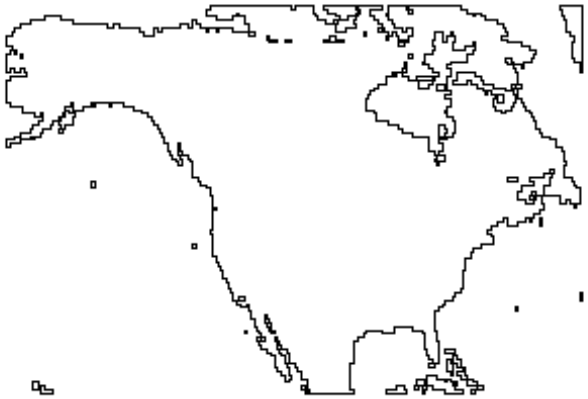
```
## Loading required package: rgdal
```

```
## 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:/Users/Chao Ma/Documents/R/win-
library/3.5/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:/Users/Chao Ma/Documents/R/win-
library/3.5/rgdal/proj
##  Linking to sp version: 1.3-1
```

```
## Loading required package: ggplot2
```

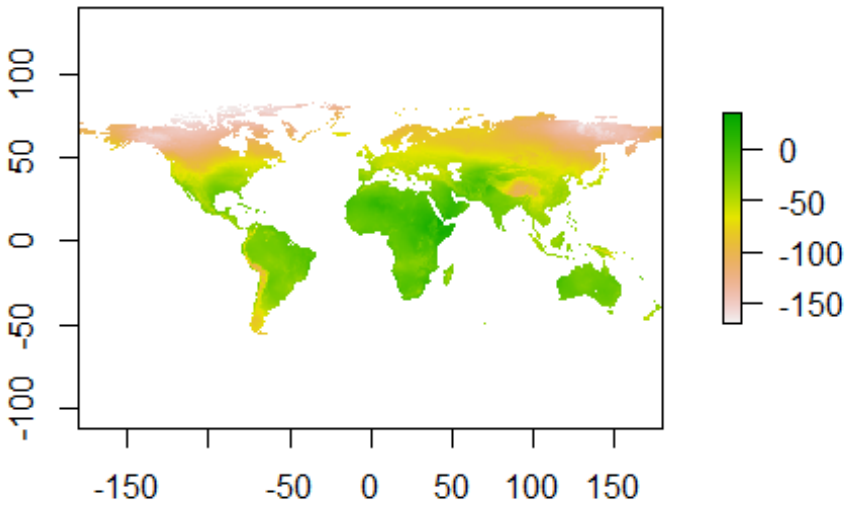
Load North America mask

```
data("naMap")
plot(naMap)
```



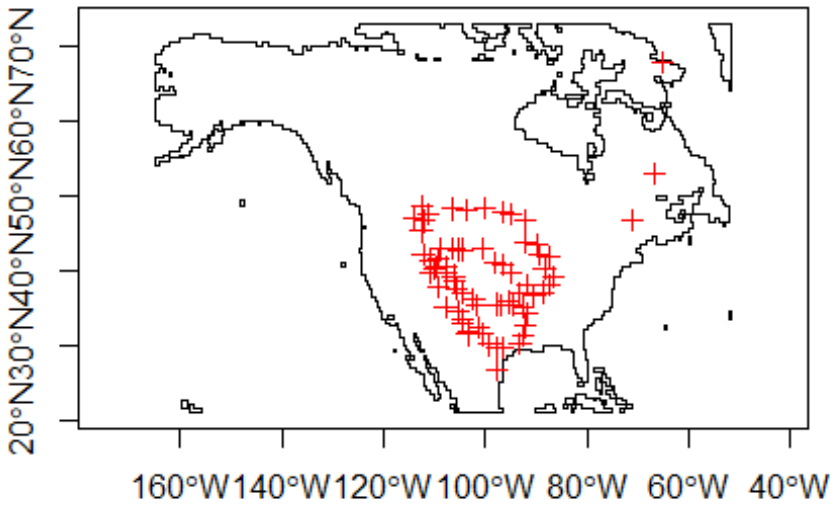
Load world precipitation hydrogen isoscape

```
data("d2h_world")
plot(d2h_world)
```



Load hydrogen isotope for human hair in North America

```
d = subOrigData(taxon = c("Homo sapiens"), mask = naMap)
```




```
## 233 data points are found
```

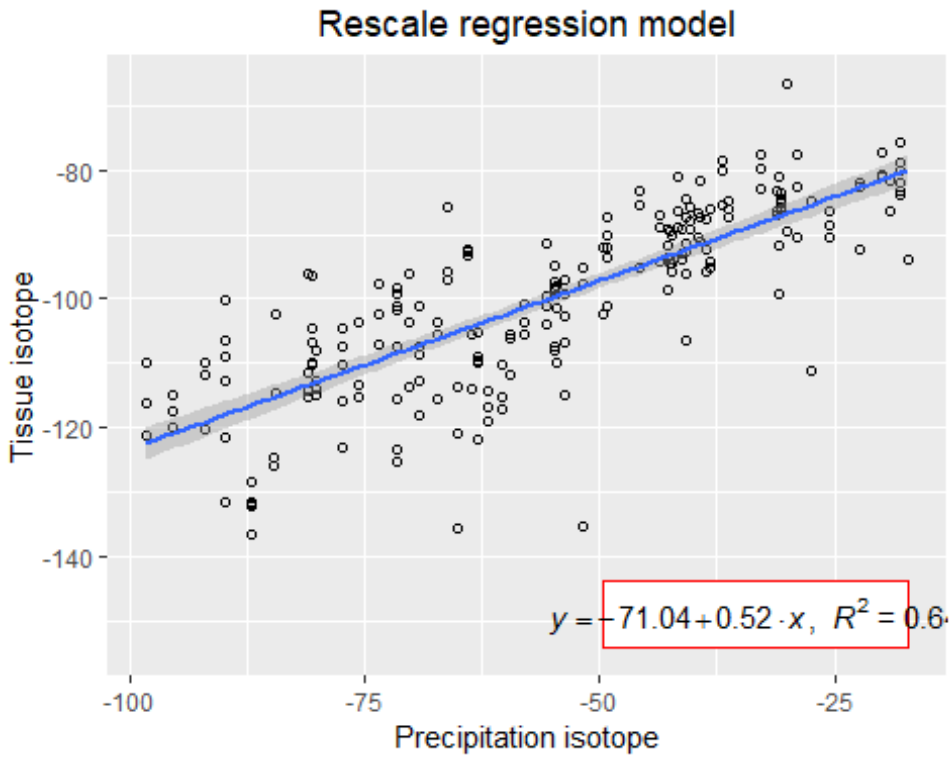
Exclude some outliers. This step is optional, which depends on your data quality

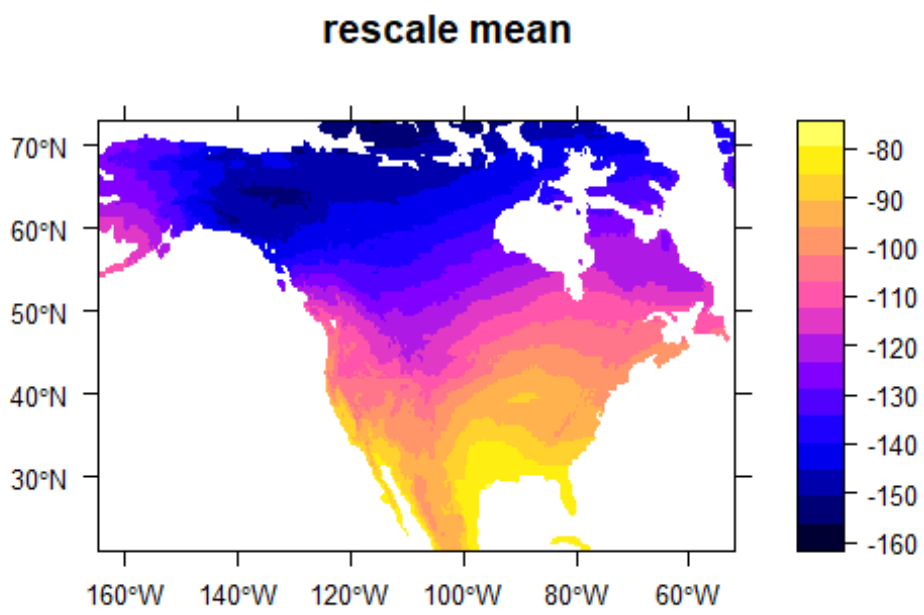
```
d <-as.data.frame(d)
dd = d[d$coords.x1<(-80),]
dd <- SpatialPointsDataFrame(dd[,2:3], as.data.frame(dd[,1]))
crs(dd) <- "+proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0"
```

Rescale from environmental isoscape to tissue isoscape

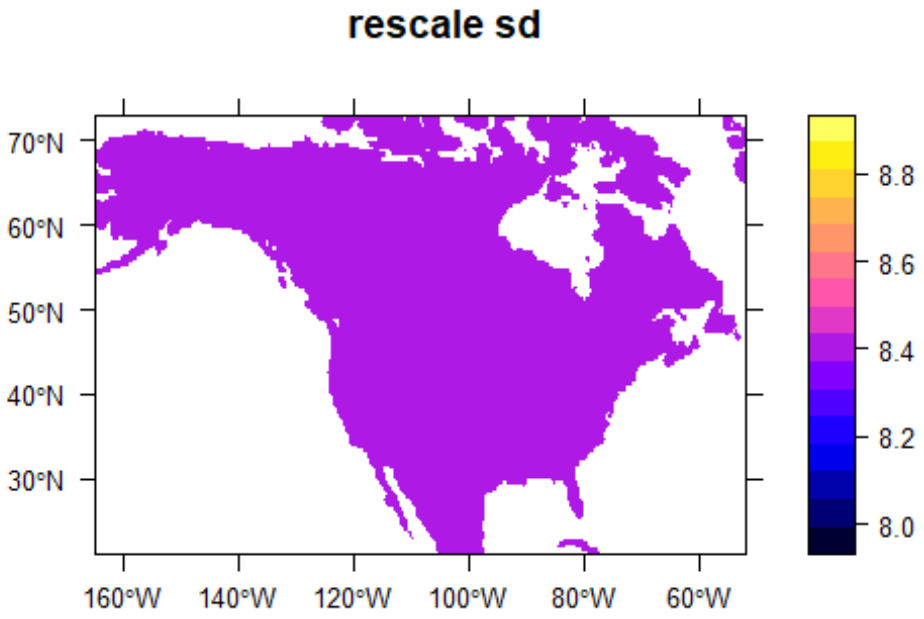
```
r = calRaster(known = dd, isoscape = d2h_world, mask = naMap)

##
##
## ------------------------------------------------------------------------
-------
## rescale function uses linear regression model, the summary of this model
is:
## ------------------------------------------------------------------------
-------
##
## Call:
## lm(formula = tissue.iso ~ isoscape.iso)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -37.317  -4.021   0.702   5.056  20.184
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -71.03824    1.59709  -44.48   <2e-16 ***
## isoscape.iso   0.52349    0.02696   19.42   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.432 on 213 degrees of freedom
## Multiple R-squared:  0.639,  Adjusted R-squared:  0.6373
## F-statistic: 377.1 on 1 and 213 DF,  p-value: < 2.2e-16
```






```
## Warning in dir.create("output"): 'output' already exists
```

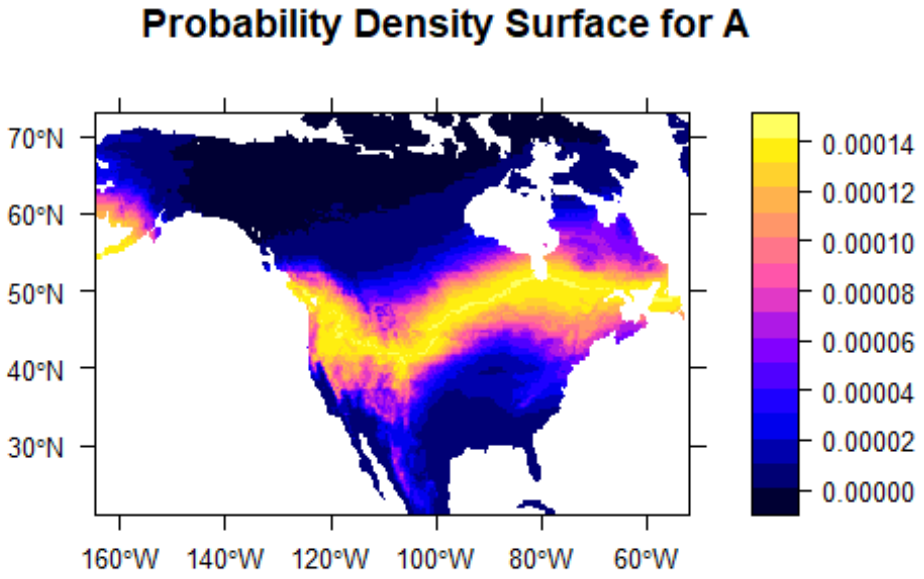


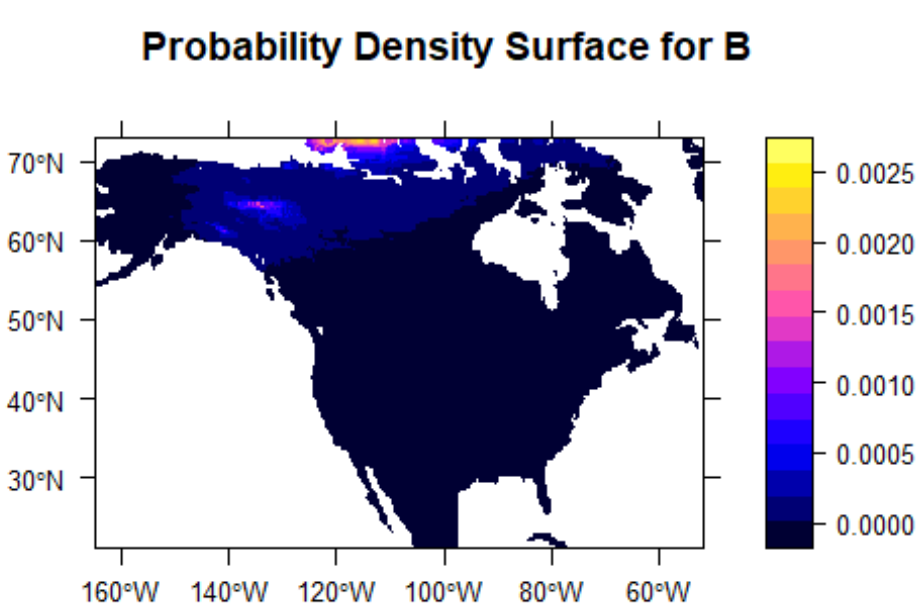
Four unknown-origin examples

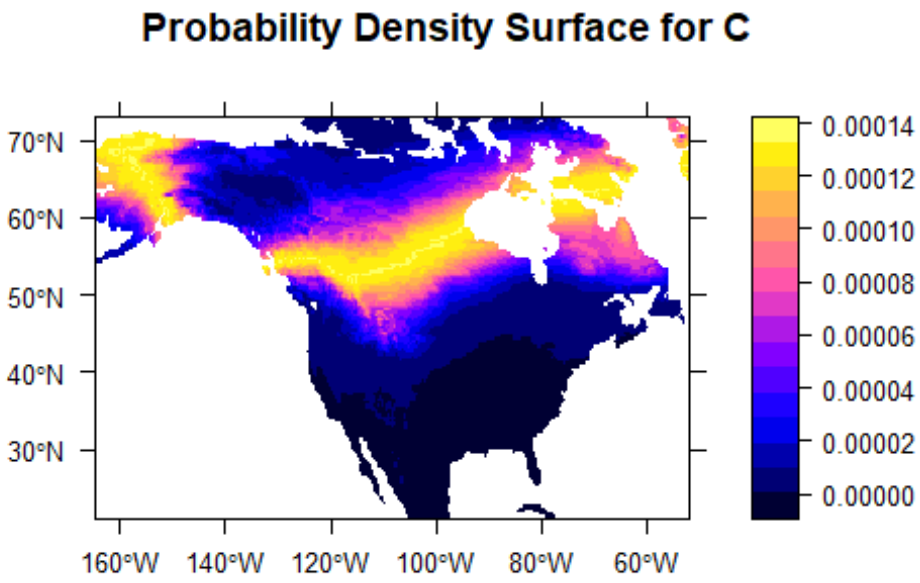
```
id = c("A", "B", "C", "D")
d2H = c(-110, -180, -130, -150)
un = data.frame(id,d2H)
```

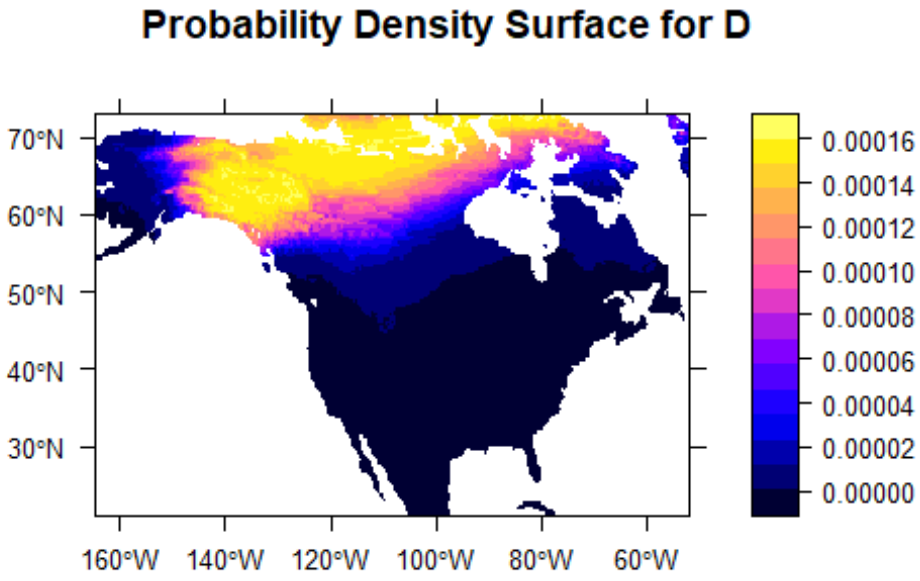
Assignment for unknown-origin examples

```
asn = pdRaster(r$isoscape.rescale,unknown=un,mask=naMap)

## Warning in dir.create("output"): 'output' already exists

## Warning in dir.create("output/pdRaster_Gtif"): 'output\pdRaster_Gtif'
## already exists
```


Probability Density Surface for A
70°N
60°N
50°N
40°N
30°N
160°W
140°W
120°W
100°W
80°W
60°W
0.00014
0.00012
0.00010
0.00008
0.00006
0.00004
0.00002
0.00000


Probability Density Surface for B
70°N
60°N
50°N
40°N
30°N
160°W
140°W
120°W
100°W
80°W
60°W
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000

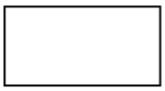




Create SpatialPolygons with two polygons

```
p1 <- c(-100,60,-100,65,-110,65,-110,60,-100,60)
p1 <- matrix(p1, 5,2, byrow = T)
p1 <- Polygon(p1)
```

```
p1 <- Polygons(list(p1), "p1")
p2 <- c(-100,40,-100,45,-110,45,-110,40,-100,40)
p2 <- matrix(p2, 5,2, byrow = T)
p2 <- Polygon(p2)
p2 <- Polygons(list(p2), "p2")
p12 <- SpatialPolygons(list(p1,p2),1:2)
plot(p12)
```



Create data.frame with two points

```
pp1 <- c(-100,45)
pp2 <- c(-100,60)
pp12 <- as.data.frame(rbind(pp1,pp2))
```

Caculate odds ratio for the two polygons created above

```
oddsRatio(asn, p12)
```

```
## $`P1/P2_odds_ratio`
##            A            B            C            D
## 1.283965e-03 3.780026e+08 3.755686e+00 6.827180e+03
##
## $`ratio of numbers of cells in two polygons`
## [1] 1
```

Caculate odds ratio for the two points created above

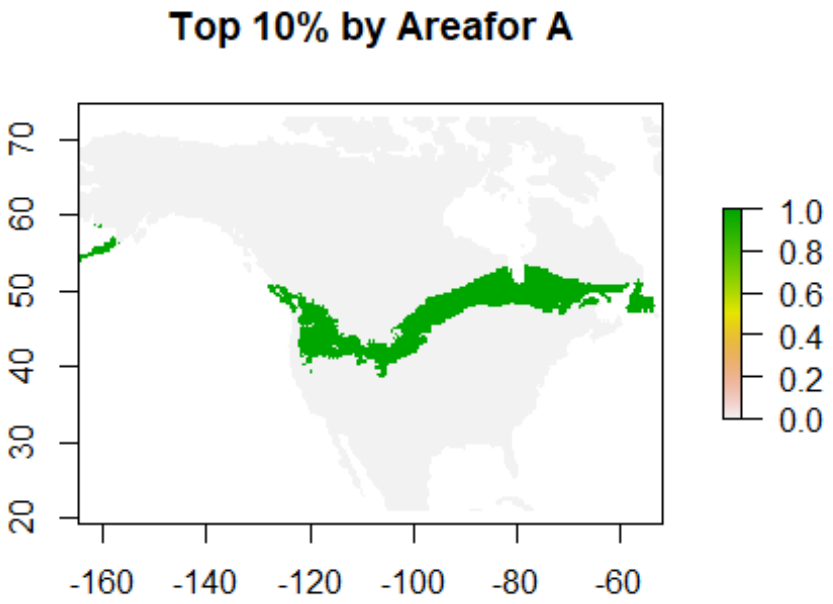
```
oddsRatio(asn, pp12)
```

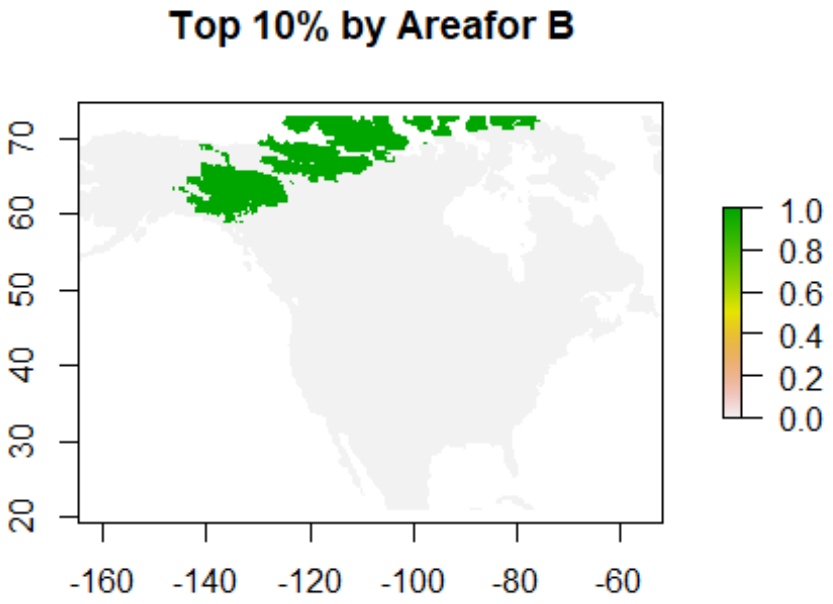
```
## $`P1/P2_odds_ratio`
##            A            B            C            D
## 9.078822e+01 2.153343e-10 4.330494e-02 2.066233e-05
##
## $`odds of a pixel to the odds of the max/min pixel`
##   ratioToMax.A ratioToMax.B ratioToMax.C ratioToMax.D ratioToMin.A
## 1  0.977370353 1.748220e-13   0.03317593 5.535731e-06 3.638540e+06
## 2  0.000593979 6.746872e-04   0.04225933 2.226383e-01 6.065034e+25
##   ratioToMin.B ratioToMin.C ratioToMin.D
## 1 9.199045e-06 1.235069e+05 2.912874e+02
## 2 8.208613e+11 4.315040e+27 2.708740e+14
```

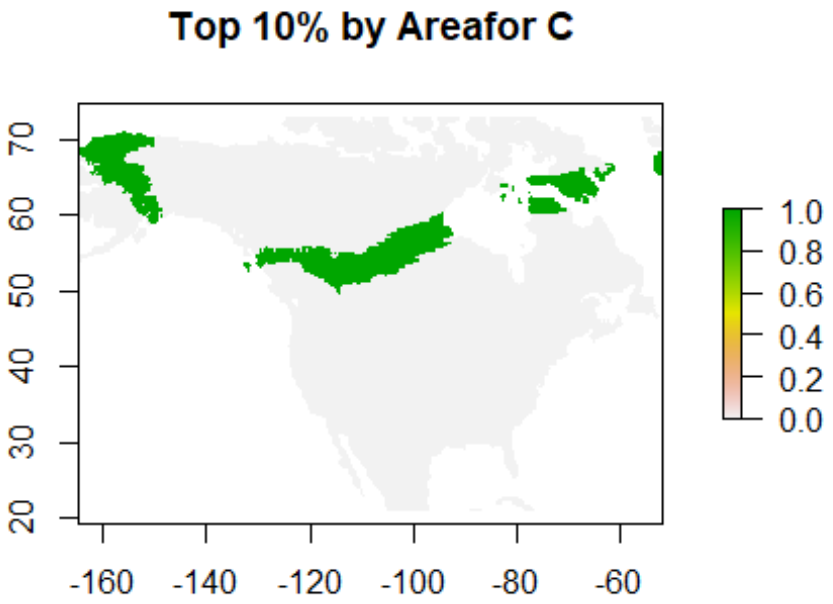
Binary reclassification

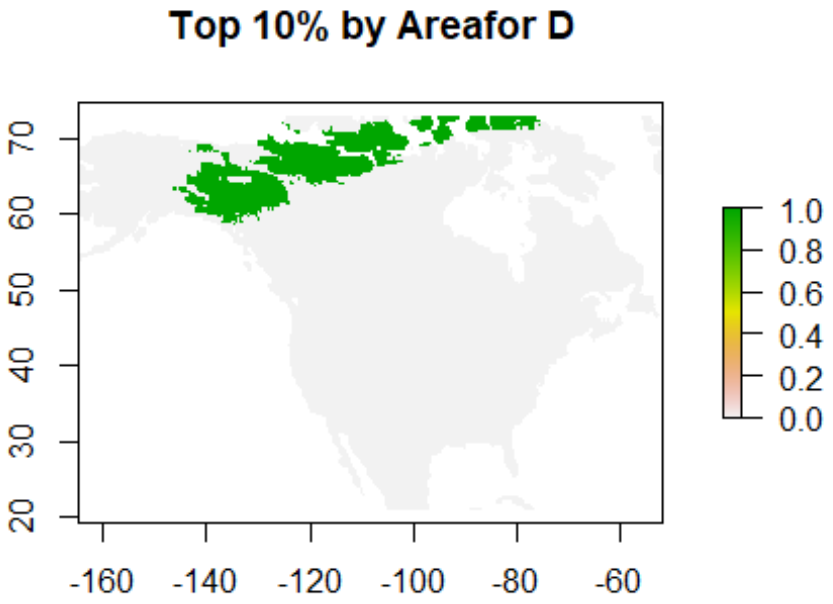
Top 10% of probability surface (defined by % area)

```
qtlRaster(asn, threshold = 0.1, thresholdType = 2)
```


Top 10% by Areafor A
70
60
50
40
30
20
-160
-140
-120
-100
-80
-60
1.0
0.8
0.6
0.4
0.2
0.0


Top 10% by Areafor B
70
60
50
40
30
20
-160
-140
-120
-100
-80
-60
1.0
0.8
0.6
0.4
0.2
0.0





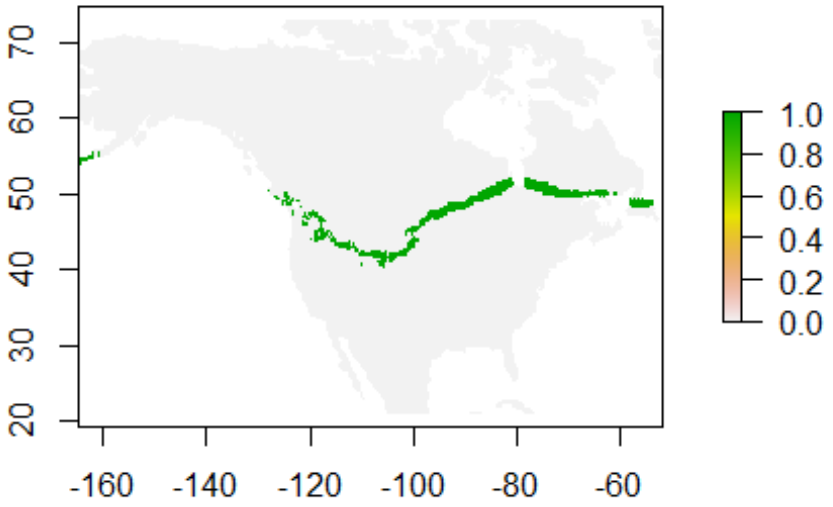

```
## class       : RasterStack
## dimensions  : 156, 339, 52884, 4  (nrow, ncol, ncell, nlayers)
## resolution  : 0.333, 0.333  (x, y)
## extent      : -164.682, -51.795, 21.06252, 73.01052  (xmin, xmax, ymin,
```

```
ymax)
## coord. ref. : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84
+towgs84=0,0,0
## names       : A, B, C, D
## min values  : 0, 0, 0, 0
## max values  : 1, 1, 1, 1
```

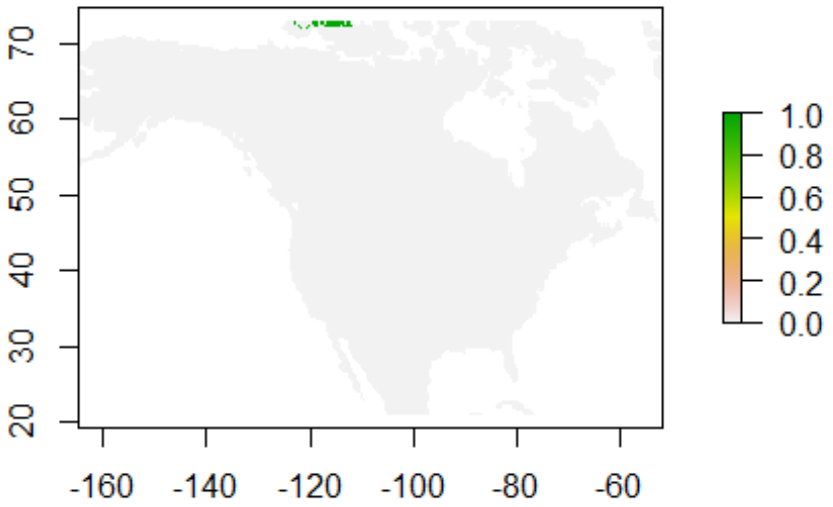
Top 10% of probability surface (defined by % cumulative probability)

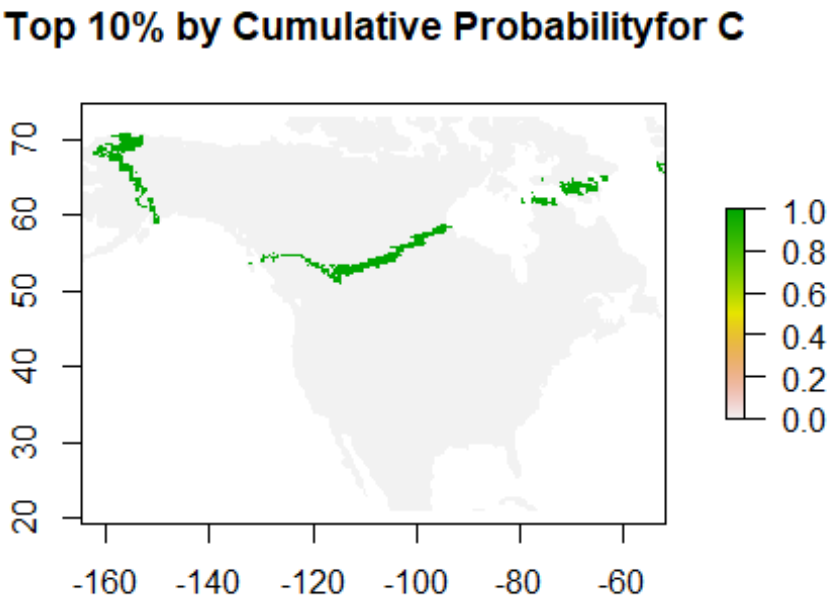
```
qtlRaster(asn, threshold = 0.1, thresholdType = 1)
```

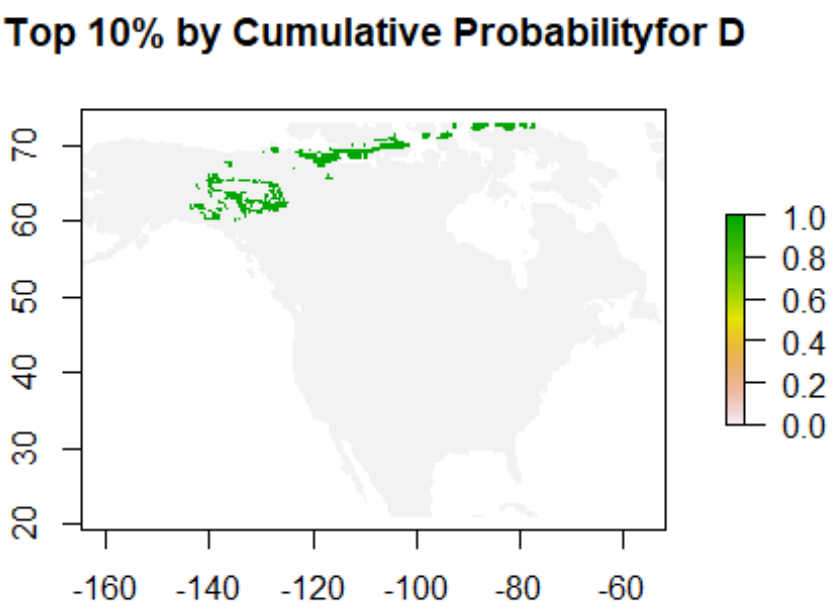
Top 10% by Cumulative Probabilityfor A



Top 10% by Cumulative Probabilityfor B





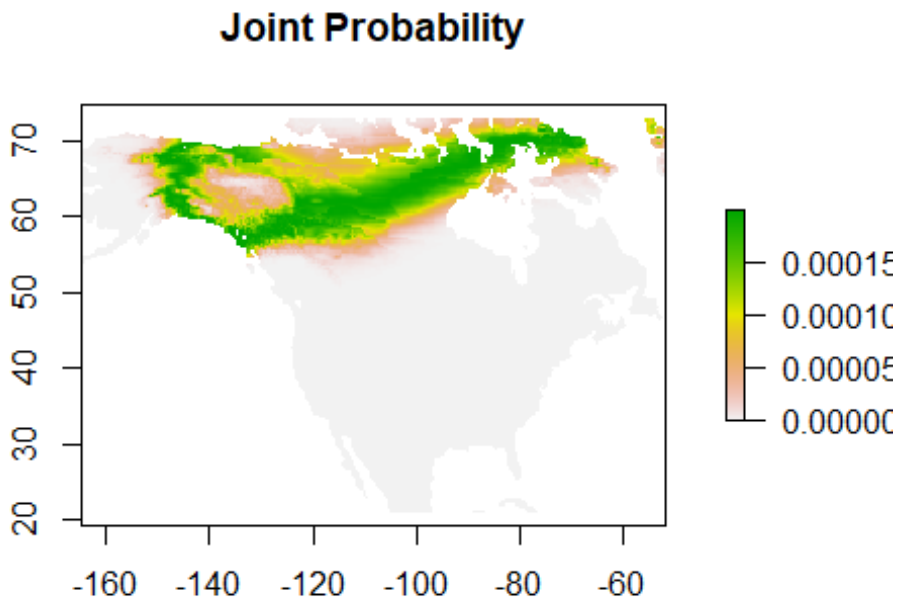



```
## class       : RasterStack
## dimensions  : 156, 339, 52884, 4  (nrow, ncol, ncell, nlayers)
## resolution  : 0.333, 0.333  (x, y)
## extent      : -164.682, -51.795, 21.06252, 73.01052  (xmin, xmax, ymin,
```

```
ymax)
## coord. ref. : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84
+towgs84=0,0,0
## names       : A, B, C, D
## min values  : 0, 0, 0, 0
## max values  : 1, 1, 1, 1
```

Joint probability for individuals of common origin

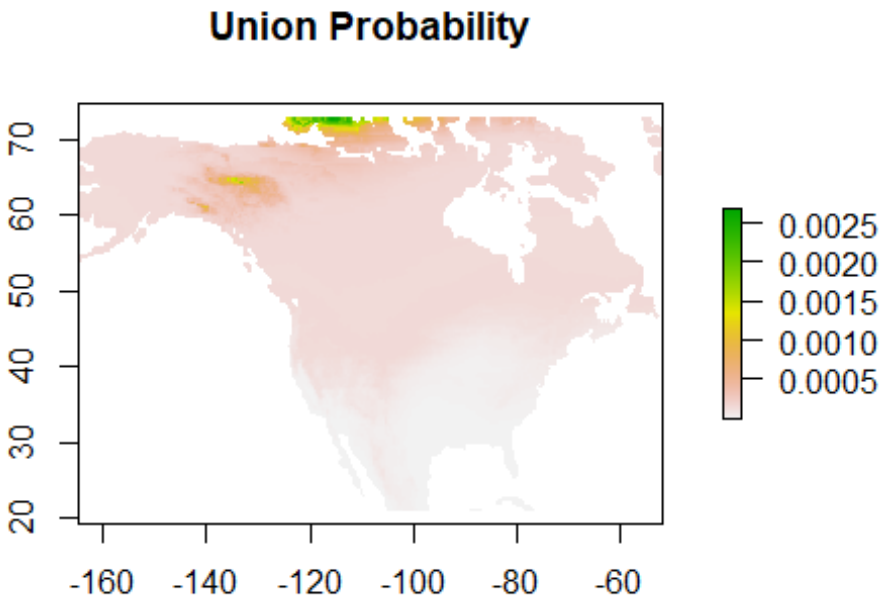
```
jointP(asn)
```




```
## class       : RasterLayer
## dimensions  : 156, 339, 52884  (nrow, ncol, ncell)
## resolution  : 0.333, 0.333  (x, y)
## extent      : -164.682, -51.795, 21.06252, 73.01052  (xmin, xmax, ymin,
ymax)
## coord. ref. : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84
+towgs84=0,0,0
## data source : in memory
## names       : Joint_Probability
## values      : 1.416425e-52, 0.0001997866  (min, max)
```

Probability that at least one individual came from the location (union of probabilities)

```
unionP(asn)
```




```
## class       : RasterLayer
## dimensions  : 156, 339, 52884  (nrow, ncol, ncell)
## resolution  : 0.333, 0.333  (x, y)
## extent      : -164.682, -51.795, 21.06252, 73.01052  (xmin, xmax, ymin,
ymax)
## coord. ref. : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84
+towgs84=0,0,0
## data source : in memory
## names       : layer
## values      : 2.239529e-07, 0.002684159  (min, max)
```