

The results below are generated from an R script.

```
# ***** FILENAME:
# Example1.R PURPOSE: Illustrate the calculation of the HR indices AUTHORS: John Fieberg
# *****;
# load kernsmooth library
library(KernSmooth) # bkde2D, dpik (plug-in estimate)

## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009

library(adehabitat) # use data from this library to illustrate functions

## Warning: package 'adehabitat' was built under R version 3.0.3
## Loading required package: ade4
##
## Attaching package: 'ade4'
##
## The following object is masked from 'package:base':
##
##   within
##
## Loading required package: tkrplot
## Loading required package: tcltk
## Loading required package: shapefiles
## Loading required package: foreign
## Warning: package 'foreign' was built under R version 3.0.2
##
## Attaching package: 'shapefiles'
##
## The following object is masked from 'package:foreign':
##
##   read.dbf, write.dbf
##
## Be careful: it is now recommended to use the newpackages adehabitatMA, adehabitatLT, adehabitatHR,
and adehabitatHS.
## These 4 packages are intended to become the future of adehabitat.
## The "classical" version of adehabitat will still be maintained for some
## time, but no new method will be added to the package.

# Change working directory (needs to point to directory holding Indices.R and calcHR.R)
# Uncomment the line below, and insert the directory name setwd('')
# Load necessary functions
source("Indices.txt")
source("CalcHR.txt")
# Data from adehabitat library (saved as a .csv file)
locs <- read.csv("puechabon_locs.csv")
animal1 <- locs[locs$Name == "Brock", c("X", "Y")]
animal2 <- locs[locs$Name == "Calou", c("X", "Y")]
# First, find reasonable axis limits Use 2 times the bandwidth to set min and max
# ranges...
htemp <- apply(rbind(animal1, animal2), 2, dpik)
min.x <- min(min(animal1$X), min(animal2$X)) - 2 * htemp[1]
max.x <- max(max(animal1$X), max(animal2$X)) + 2 * htemp[1]
min.y <- min(min(animal1$Y), min(animal2$Y)) - 2 * htemp[2]
max.y <- max(max(animal1$Y), max(animal2$Y)) + 2 * htemp[2]
```

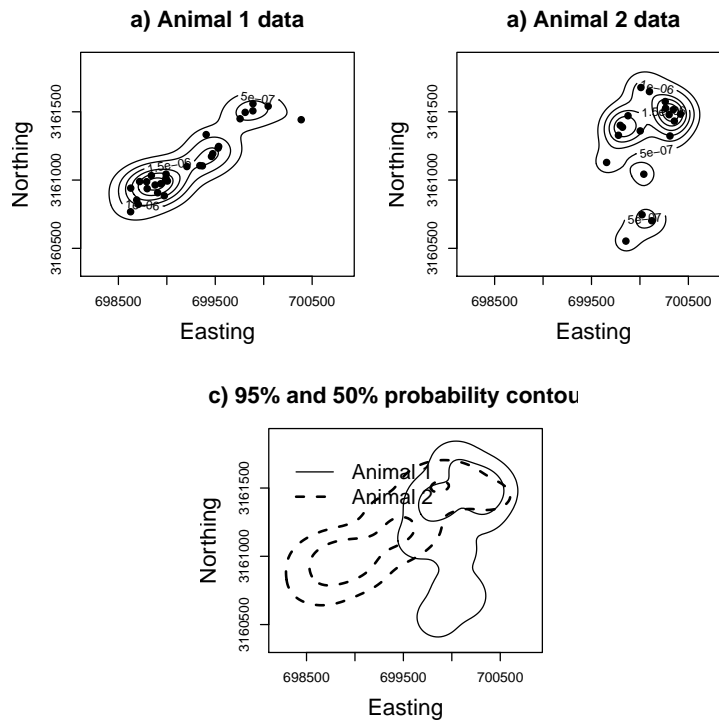
```

# Calculate Home ranges with plug in method (separate bandwidths in x and y directions)
UD1 <- bkde2D(animal1, apply(animal1, 2, dpik), gridsize = c(101, 101), range.x = list(c(min.x,
max.x), c(min.y, max.y)))
UD2 <- bkde2D(animal2, apply(animal2, 2, dpik), gridsize = c(101, 101), range.x = list(c(min.x,
max.x), c(min.y, max.y)))
# Now, get 50% and 95% contour info using function calcHR Returns: 1. HR = HR size (for
# each contour level; default= 50%, 95%) 2. ps = probability level associated with the pth
# probability contour 3. totp = total probability level (should be close to 1 if grid size
# is adequate)
animal1hr <- calcHR(UD1)
animal2hr <- calcHR(UD2)
# Get grid cell size
dxdy <- (UD1$x1[2] - UD1$x1[1]) * (UD1$x2[2] - UD1$x2[1])
# Get 50% and 95% conditional distributions
UD1.50 <- UD1$fhat
UD1.50[UD1.50 < animal1hr$ps[1]] <- 0
UD1.50 <- UD1.50/sum(UD1.50 * dxdy)
UD2.50 <- UD2$fhat
UD2.50[UD2.50 < animal2hr$ps[1]] <- 0
UD2.50 <- UD2.50/sum(UD2.50 * dxdy)
UD1.95 <- UD1$fhat
UD1.95[UD1.95 < animal1hr$ps[2]] <- 0
UD1.95 <- UD1.95/sum(UD1.95 * dxdy)
UD2.95 <- UD2$fhat
UD2.95[UD2.95 < animal2hr$ps[2]] <- 0
UD2.95 <- UD2.95/sum(UD2.95 * dxdy)
# Calculate overlap indices for 50% and 95% conditional distributions
resultsmat <- matrix(0, 2, 7)
resultsmat[1, ] <- allcalc(UD1.50, UD2.50, dxdy)
resultsmat[2, ] <- allcalc(UD1.95, UD2.95, dxdy)
colnames(resultsmat) <- c("VI", "HR1,2", "HR2,1", "PHR1,2", "PHR2,1", "BA", "UDOI")
rownames(resultsmat) <- c("50%", "95%")
round(resultsmat, 2)

##          VI HR1,2 HR2,1 PHR1,2 PHR2,1  BA UDOI
## 50% 0.03  0.04  0.04   0.03   0.03 0.03 0.00
## 95% 0.23  0.39  0.41   0.62   0.28 0.37 0.16

# Figure 1 (3 panel plot) a) = UD1 data (points & contours) b) = UD2 data (points &
# contours) c) = 50% and 95% contours overlaid
# Set up plotting region
par(cex.lab = 1.7, cex.main = 1.7)
layout(matrix(c(1, 1, 2, 2, 0, 3, 3, 0), 2, 4, byrow = TRUE))
# Panel a.
contour(UD1$x1, UD1$x2, UD1$fhat, xlab = "Easting", ylab = "Northing", main = "a) Animal 1 data")
points(animal1$X, animal1$Y, pch = 20, cex = 1.5)
# Panel b
contour(UD2$x1, UD2$x2, UD2$fhat, xlab = "Easting", ylab = "Northing", main = "a) Animal 2 data")
points(animal2$X, animal2$Y, pch = 20, cex = 1.5)
# Panel c
contour(UD1$x1, UD1$x2, UD1$fhat, levels = animal1hr$ps, xlab = "Easting", ylab = "Northing",
main = "c) 95% and 50% probability contours", lty = 2, lwd = 2, drawlabels = F)
contour(UD2$x1, UD2$x2, UD2$fhat, add = T, levels = animal2hr$ps, drawlabels = F)
legend(698200, 3161800, lty = c(1, 2), lwd = c(1, 2), c("Animal 1", "Animal 2"), cex = 1.5,
bty = "n")

```



The R session information (including the OS info, R version and all packages used):

```

sessionInfo()

## R version 3.0.1 (2013-05-16)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
##
## locale:
## [1] LC_COLLATE=English_United States.1252 LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252 LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] tcltk      stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
## [1] adehabitat_1.8.12  shapefiles_0.7    foreign_0.8-57    tkrplot_0.0-23
## [5] ade4_1.5-2        KernSmooth_2.23-10  knitr_1.4.1
##
## loaded via a namespace (and not attached):
## [1] digest_0.6.3    evaluate_0.4.7    formatR_0.9      highr_0.2.1     stringr_0.6.2
## [6] tools_3.0.1

Sys.time()

## [1] "2014-04-18 17:04:05 CDT"

```