

Appendix 3: Home Range Estimation using AKDE Walkthrough

Odd Jacobson

2023-01-26

The purpose of this appendix is to detail the steps to estimating a home range using the *ctmm* package. Accounting for autocorrelation is important so that we avoid biases in our results. However, it requires some additional steps compared to most conventional estimators, which is why we describe the process and provide an example workflow. We strongly recommend going through the *ctmm* vignettes (see <https://ctmm-initiative.github.io/ctmm/index.html>) for a detailed review.

Generating a home range estimate from movement data using continuous-time movement modelling involves three main steps: 1) Variogram inspection, 2) Model fitting and selection, and 3) AKDE calculations. This process can either be done using the *ctmm* package in the R environment for statistical computing, or using the *ctmmweb* point-and-click graphical user interface (Calabrese et al., 2021), which streamlines the modelling steps, helping users conduct home range analysis without the need to know the R programming language. We describe the former method below:

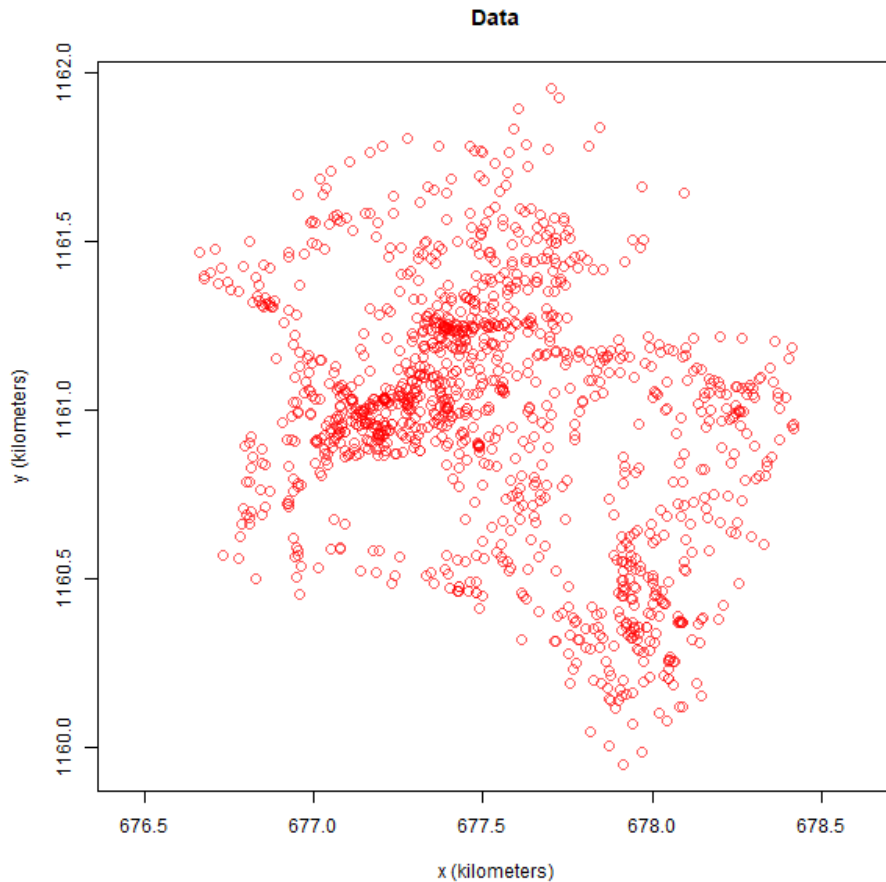
The first step is to load the necessary packages and read in the data

```
library(tidyverse)
library(ctmm)
library(sf)
library(ctmmweb)
```

Data must have the same format as the following data frame. These are the same format required by *Movebank*. Either you can manually edit the dataframe and then convert to a telemetry object, or put data on Movebank and read data in from there, which will automatically put the data in the correct format. Once data is in correct format, convert to telemetry object and specify the UTM projection.

```
# read in data frame
DATA <- read.csv("Data/CH1_GPS_data.csv", row.names = NULL) %>%
  filter(group == "SP" & individual.local.identifier == "all") %>% #select the complete
  ↪ segment from SP group (could pick any)
  as.telemetry(projection = "+proj=utm +zone=16 +north +datum=WGS84 +units=m +no_defs
  ↪ +ellps=WGS84 +towgs84=0,0,0") # change to telemetry object for ctmm

plot(DATA, main = "Data")
```

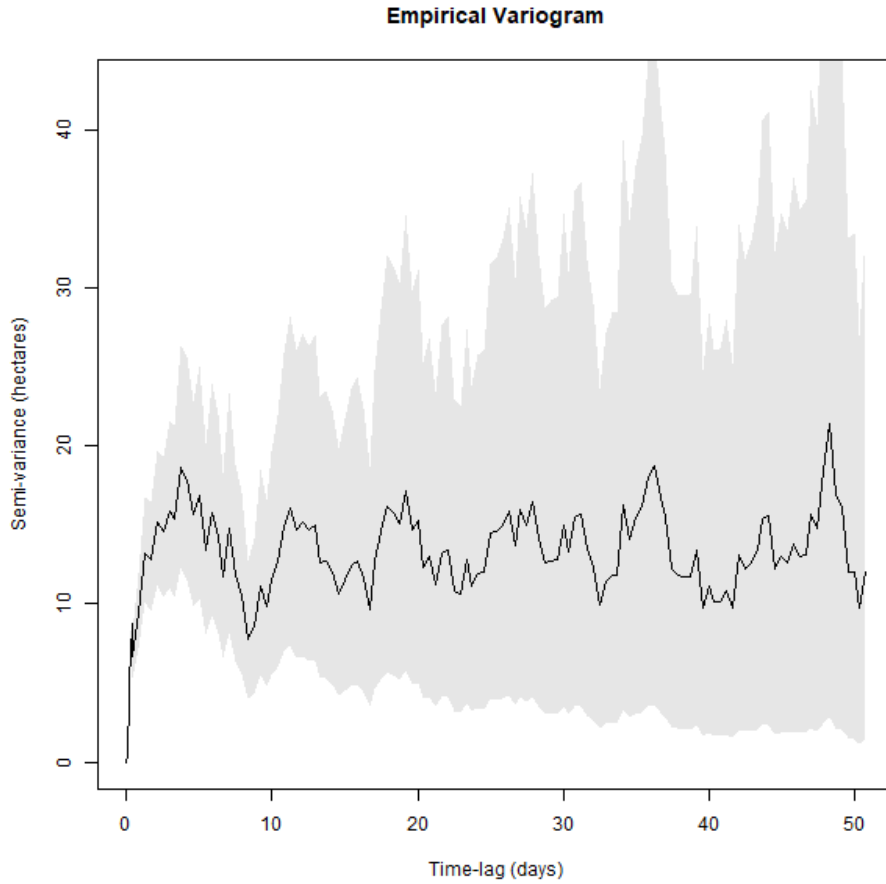


Varigram Inspection

Variograms plot the semi-variance (y-axis), which is a measure of the average squared displacement, as a function of the time-lag that separates any pair of observed locations (Diggle and Ribeiro 2007; Silva et al. 2021). Variograms play two major roles in the *ctmm* workflow: first, they provide an unbiased visual diagnostic to assess the autocorrelation structure present in the data, and second, they inform whether the data shows evidence of range residency (Silva et al. 2021). Asymptoting curves in a variogram indicate range residency. Where the asymptote aligns with the x-axis is a measure of the necessary time-lag between positions to assume independence (Silva et al. 2021). It is also a rough estimate of the home range crossing time (Christen H. Fleming and Calabrese 2017). If the curve continues to increase without flattening, the animals are either non-resident (i.e. home range drift or migration), or not tracked long enough to capture the full extent of their home ranges (Calabrese, Fleming, and Gurarie 2016).

We used the *variogram* and *plot* functions to visually assess the empirical variogram curves. Once we confirmed that all complete segments represented range-restricted movement, we proceeded with model fitting and selection. It is necessary to confirm range-residency before conducting home range estimation because, while *ctmm* is capable of fitting both range-resident (the default) and endlessly diffusing movement models, only the first set are appropriate for home range estimation.

```
SVF <- variogram(DATA, dt = c(1,10) %%% "hour") # dt argument changes the width of the
  ↪ time-lag bins (makes variogram smoother)
plot(SVF, main = "Empirical Variogram")
```



This is a plot of the empirical variogram. The line asymptotes at approximately five days. This is roughly the average home range crossing time. This is also approximately how far apart locations need to be in time for them to be independent (<https://ctmm-initiative.github.io/ctmm/articles/variogram.html>).

Movement Model Selection

Again, see the above link for a detailed review. We will only briefly cover the necessary steps, but to fully understand the model selection process it is important to through the *ctmm* vignette.

Model fitting also involves two steps: first the *ctmm.guess* function uses the shape of the empirical variogram to generate starting values required for the non-linear models, and second, the *ctmm.select* function uses the values calculated from *ctmm.guess* to fit a range of alternative stationary (and range-restricted) movement models using Maximum Likelihood (C. H. Fleming et al., 2014). Models are ranked by AICc (Akaike information criterion) allowing us to evaluate which model or models best predict the data. This process permits identification and fit of a stationary movement model that corresponds to the observed movement behavior of the animal (C. Fleming et al. 2014).

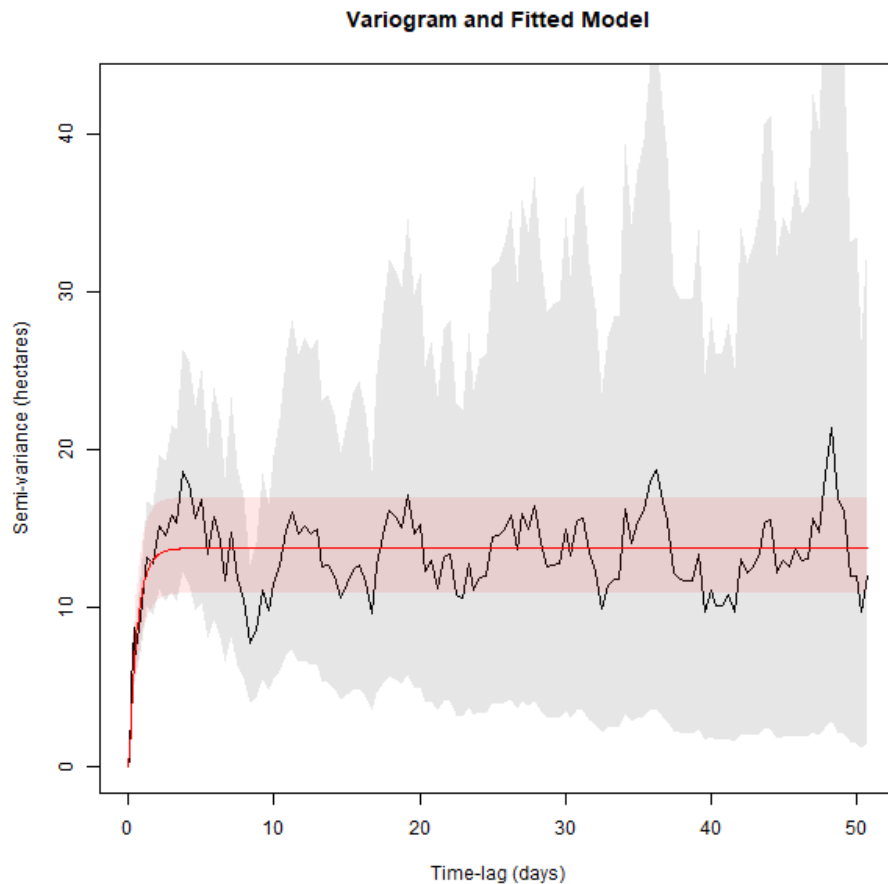
It is worth noting that here, stationary means that the underlying movement processes are assumed to be consistent throughout the duration of the data. Movement model parameters represent time-averaged values, which has important implications on how data should be segmented for home range analysis. If the underlying parameters change drastically within the sample—particularly the mean location—then the stationary assumption has been violated. Therefore, it is common practice to segment the data when the parameters change and estimate separate ranges. This is consistent with Burt’s original concept of the home range where, for example, he stated winter and summer ranges for migratory species should be considered

separately with the travel between as transit (Burt, 1943). In our case, all subsets were from single, stationary ranges which negated any need for further segmentation.

```
GUESS <- ctm.guess(DATA, interactive=FALSE, variogram = SVF)
FIT <- ctm.select(DATA, GUESS, trace=2)

# model summary
summary(FIT)

# plot empirical variogram with best model
plot(SVF, CTMM = FIT, main = "Variogram and Fitted Model")
```



The fitted model provide estimates and confidence intervals for several related quantities, including mean speed (km/day i.e. proportional to daily travel distance), diffusion rate, position autocorrelation timescale or the home range crossing time ($\tau[\text{position}]$ (hours)), velocity autocorrelation timescale ($\tau[\text{velocity}]$ (minutes)), and the effective sample size or number of home range crossings in the data (i.e. “\$DOF area”).

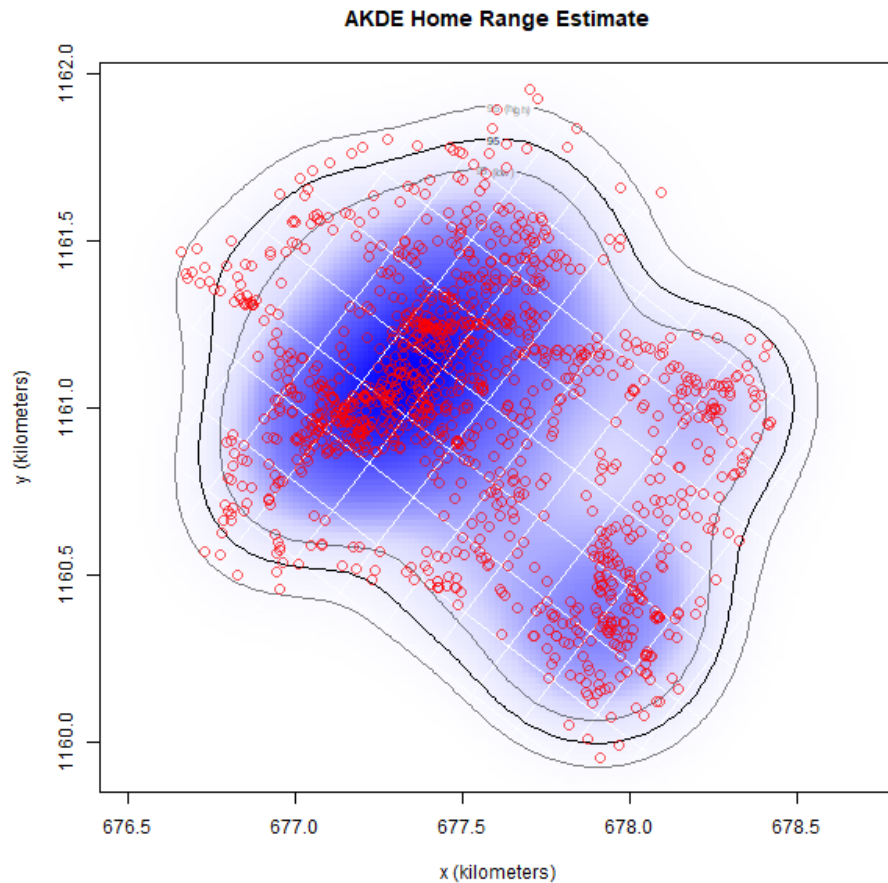
AKDE Home Range Estimation

The final stage is to calculate an autocorrelated kernel density home-range estimate (AKDE) using the eponymously named *akde* function (Calabrese, Fleming, and Gurarie 2016). This function takes the movement data and the corresponding fitted model and returns: a UD object corresponding to the range distribution,

information on the optimal bandwidth, point estimates and confidence intervals for area, and a measure of the effective sample size of the data for home range estimation. In our case, we selected *weights = TRUE* within the *akde* function, which helps correct for irregular and missing data by down-weighting over-sampled portions of the data and up-weighting under-sampled portions (C. H. Fleming et al. 2018). This helps to offset sampling bias, but is not sufficient if large portions of the true range are missing from the sampled data.

```
UD <- akde(DATA, FIT, weights = TRUE)

# plot UD
plot(DATA, UD = UD, main = "AKDE Home Range Estimate")
```



References