SCHOOL OF INFORMATION SCIENCES & TECHNOLOGY

DEPARTMENT OF STATISTICS
POSTGRADUATE PROGRAM

Home range analysis of real telemetry data of a greylag goose inhabiting the Prespa lakes, northwestern Greece.

By
Christina K. Boleti

A THESIS
Submitted to the Department of Statistics
of the Athens University of Economics and Business
in partial fulfilment of the requirements for
the degree of Master of Science in Statistics

Athens, Greece
March 2015

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

ΣΧΟΛΗ ΕΠΙΣΤΗΜΩΝ & ΤΕΧΝΟΛΟΓΙΑΣ ΤΗΣ ΠΛΗΡΟΦΟΡΙΑΣ

ΤΜΗΜΑ ΣΤΑΤΙΣΤΙΚΗΣ

ΜΕΤΑΠΤΥΧΙΑΚΟ

Ανάλυση εύρους κατοικίας απο πραγματικά GPS δεδομένα μίας σταχτοχήνας που ενδημεί στις λίμνες των Πρεσπών, βορειοδυτική Ελλάδα

Χριστίνα Κ. Μπολέτη

ΔΙΑΤΡΙΒΗ

Που υποβλήθηκε στο Τμήμα Στατιστικής

του Οικονομικού Πανεπιστημίου Αθηνών

ως μέρος των απαιτήσεων για την απόκτηση

Μεταπτυχιακού Διπλώματος Ειδίκευσης στη Στατιστική

Αθήνα

Μάρτιος 2015

ACKNOWLEDGEMENTS

I would wish to acknowledge the contribution made by the following in preparing this thesis:

Panagiotis Besbeas (Assistant Professor, Athens University of Economics and Business, Department of Statistics.) besbeas@aueb.gr
Giorgos Catsadorakis (PhD, Conservation Biologist, Scientific Advisor to the Society for the Protection of Prespa.) doncats@otenet.gr
Annita Logotheti (Environmental scientist, Conservation officer at the Society for the Protection of Prespa.) a.logotheti@spp.gr
Olga Alexandrou (Ornithologist-Forester, Conservation officer at Society for the Protection of Prespa.) o.alexandrou@spp.gr
Society for the Protection of Prespa, www.spp.gr

II
I

VITA

I was born in Larisa where I lived until I was 18 years old. In 2006 I entered the department of Mathematics at the National and Kapodistrian University of Athens. In 2013 I enrolled at the Master program of statistics in Athens University of Economics and Bussiness. My thesis was related to environmental statistics, which I find a fascinating area and I would like to practice it more in depth in the future.

ABSTRACT

Animal movement data derived from satellite telemetry have shown a considerable increase in recent years due to technological advances in the field. Global positioning system (GPS) is the most modern technique for determining movements, home range and habitat use of animals in the wild. Miniature transmitters housing a GPS are fitted to study animals and communicate locational information of these individuals via various means such as satellites, GSM network, blue-tooth technology, depending on the type of the transmitter. We examine the use of GPS methods to estimate the home range of a greylag goose (*Anser anser rubrirostirs*) in Prespa, Greece. Our goal was not to provide a complete comparison of all methods, but rather to use freely available methods and software to highlight the challenges of estimating home range with large GPS datasets. Our review detailed proposed methods to use on autocorrelated locations that are common in GPS datasets and explain the abilities of software, or lack of theory, to calculate home range of animals. Specifically we focused on: the Kernel Density Estimation method (KDE), the Brownian Bridge Movement Method (BBMM) and its improvement, Dynamic Brownian Bridge Movement Method (dBBMM) as well as, comparison of these methods in order to estimate the home range of the greylag goose. Furthermore, we methodologically extend the Brownian Bridge Movement Method to estimate an additional variance parameter typically assumed known in applications. The variability in time lag between successive locations was important in our data-set due to solar-powered GPS and the BBMM method resulted in producing larger home range size than the KDE with reference bandwidth, which fails to take into consideration the time lag. In addition, dBBMM slightly outperformed the BBMM and the home range polygons generated with plug-in bandwidth appeared fragmented and possibly underestimated the home range.

VI
I

Contents

Chapter 1

Introduction

1.1 Introduction

Recent advances in global positioning system (GPS) technology for monitoring home range and movements of wildlife have resulted in locations that are numerous, more precise than very high frequency (VHF) systems, and often are autocorrelated in space and time (Walter 2011).

The typical approach to analyse and visualize the area used by a tracked animal is to convert its movement into a 2-dimensional spatial representation originally referred to as a **home range** (Burt 1943). Home range is defined as *"that area traversed by the animal during its normal activities of food gathering, mating and caring for young. Occasional sallies outside the area, perhaps exploratory in nature, should not be considered as in part of the home range"* (Burt 1943). The need for performing statistical analyses of home ranges has led to more explicit definitions such as **utilization distribution** (UD), defined as *"the two dimensional relative frequency distribution for the points of location of an animal over a period of time"* (Van Winkle 1975). Thus, the utilization distribution is a probabilistic model of home range that describes the relative amount of time that an animal spends in a place. Within such a framework one can then define home range as "the smallest sub-region which accounts for a specified proportion of its total utlization" (Jennrich & Turner 1969).

The first methods used to estimate UDs assumed simple bivariate normal models (Calhoun & Casby 1958, Jennrich & Turner 1969). Don & Rennolls (1983) used a mixture of bivariate normal distributions that allowed for the possibility of multimodality. It is possible to think of other, more sophisticated models that could be used to describe the UD. However, it is also worth keeping models as simple as possible, but still providing a reasonable fit to the data. Because of the processes that give rise to home range area usage, it is not difficult to imagine

that UDs might arise that would be difficult to model using standard bivariate distributions (Dixon & Chapman 1980, Anderson 1982). For this reason, when a simple parametric model is inappropriate or difficult to specify, there is a need for nonparametric estimation approaches (Worton 1989).

In recent years UDs are commonly estimated with *kernel methods* using a collection of spatial points that ignore the temporal structure (Worton 1989), requiring individual points to be either sampled from a track at regular intervals or temporally independent (Fieberg 2007; Fieberg 2010). However, kernel methods have not been useful for modern GPS data sets because of violation of independence assumptions of the location data and for errors in proper bandwidth selection. Thus, there is a need to develop new UD methods that can accommodate the more detailed animal tracks provided by modern GPS tracking (Kie 2010).

The recent introduction of the *Brownian bridge movement model* (BBMM) improves on the traditional UD statistics by incorporating the temporal structure of tracking data and explicitly modelling the movement path (Bullard 1999; Horne 2007). The BBMM does this by incorporating both the order of locations and the amount of time between them. The model approximates the movement path between two subsequent locations by applying a conditional random walk. The BBMM has been rapidly adopted because it provides straightforward results, it is based on clear assumptions, it can incorporate location errors and it is simple to apply to a wide range of movements (Lonergan, Fedak & McConnell 2009; Ovaskainen & Crone 2009; Willems & Hill 2009). Consequently, the BBMM has been recognized for its broad potential in ecological studies, for example, to calculate encounter rates of animals (e.g. Farmer 2010) or model disease outbreaks (Takekawa 2010).

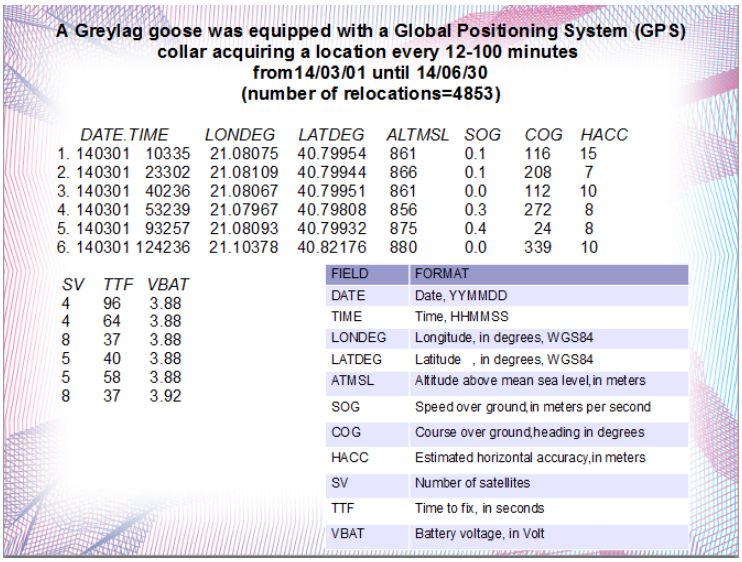
However, the BBMM has been criticised in the literature as it assumes unrealistic homogeneous movement behavior across all data. In particular, the current BBMM assumes animal movement patterns within a track to follow one constant property. Kranstauber (2012), based on a previous method *Behavioural Change Point Analysis* (Gurarie, Andrews & Laidre 2009), improve the BBMM by allowing for changes in behavior, using likelihood statistics to determine change points along the animal's movement path. The *dynamic Brownian Bridge Movement Method* (dBBMM) proposed by Kranstauber (2012) performs better than or at least similar (in cases that tracks have low variation in movement patterns) to the BBMM, but never worse.

1.2 Application to data

We applied the methods to estimate home range to global positioning system (GPS) data collected from a male **Greylag goose** ***(Anser anser rubrirostirs)*** tagged in spring 2014 in Prespa, Greece. The male was fitted with a neckband solar-powered GPS collar with high accuracy, but produced large data intervals because of failure to charge the battery. The animal was monitored for four months (March-June), during which no large trip occurred. A total of 4852 position fixes were obtained during this period. The time intervals between the fixes ranged widely from less than 13 minutes to 300 minutes, with the majority (over 80%) between 12 and 100 min. Thus, the trajectory is irregular, since it is characterized by a variable time lag between successive relocations. The first six observations are shown in Figure 1.1.

Although the use of the geographic coordinate system (i.e. latitude, longitude) is recommended in cases of long distance movements and is often the default geographic collection method for GPS collar data, some home range software (e.g. `adehabitatHR` package in R) requires input coordinate data to be in meters. This is challenging when global positioning system technology has been used to document movements of wildlife that migrate long distances (Mandel 2008; Sawyer 2009; Takekawa 2010). With such movements, an animal could occupy more than 5 Universal Transverse Mercator (UTM) zones during migrations from southern to northern latitudes (Walter 2011). Therefore, the home range analysis of this animal might be best depicted using Albers Equal Area or Lambert Conformal. Home range can be estimated for many animals within their respective UTM zone if GPS locations do not extend outside of more than one zone (see Appendix B).

Geographic positions were converted to displacements in meters, using the UTM projection, *zone 34N (WGS84)* for estimating home range except for the last method (dBBMM), implemented in the `move` package in R, that required the coordinates to be in Azimulthal Equidistant projection (see Appendix C).



A Greylag goose was equipped with a Global Positioning System (GPS) collar acquiring a location every 12-100 minutes from14/03/01 until 14/06/30 (number of relocations=4853)

	DATE	TIME	LONDEG	LATDEG	ALTMSL	SOG	COG	HACC
1.	140301	10335	21.08075	40.79954	861	0.1	116	15
2.	140301	23302	21.08109	40.79944	866	0.1	208	7
3.	140301	40236	21.08067	40.79951	861	0.0	112	10
4.	140301	53239	21.07967	40.79808	856	0.3	272	8
5.	140301	93257	21.08093	40.79932	875	0.4	24	8
6.	140301	124236	21.10378	40.82176	880	0.0	339	10

SV	TTF	VBAT
4	96	3.88
4	64	3.88
8	37	3.88
5	40	3.88
5	58	3.88
8	37	3.92

FIELD	FORMAT
DATE	Date, YYMMDD
TIME	Time, HHMMSS
LONDEG	Longitude, in degrees, WGS84
LATDEG	Latitude , in degrees, WGS84
ATMSL	Altitude above mean sea level,in meters
SOG	Speed over ground,in meters per second
COG	Course over ground,heading in degrees
HACC	Estimated horizontal accuracy,in meters
SV	Number of satellites
TTF	Time to fix, in seconds
VBAT	Battery voltage, in Volt

Figure 1.1: *The first six relocations of the greylag goose from a total of 4852 relocations.*

In the figure below we can see the number of locations (4852), the extent of the coordinates (map extent), the coordinate reference system (or projection, here longitude - latitude), the number of columns of the imported data.frame, the names of the columns of the data.frame and their minimum and maximum values, as well as the first and last timestamp and the duration of the observation.

```
features   : 4852
extent     : 21.06339, 21.12755, 40.78646, 40.82802  (xmin, xmax, ymin, ymax)
coord. ref. : +proj=longlat +ellps=WGS84 +datum=WGS84 +towgs84=0,0,0
variables  : 11
names      :   DATE.TIME,  LONDEG,  LATDEG, ALTMSL,  SOG, COG, HACC, SV, TTF,
VBAT, sensor
min values : 140301 010335, 21.06339, 40.78646,   -10,  0.0,   0,   3,  4,  37, 3.86,   gps
max values : 140630 225237, 21.12755, 40.82802,   949, 20.8, 359,  99, 12,  97, 4.12,   gps
timestamps : 2014-03-01 01:03:35 ... 2014-06-30 22:52:37 Time difference of 122 days  (start ...
end, duration)
sensors    : gps
indiv. data : individual.local.identifier
indiv. value: X1
date created: 2014-11-13 01:40:26
```

A graphical display of the distance between two successive relocations of the goose and the time:

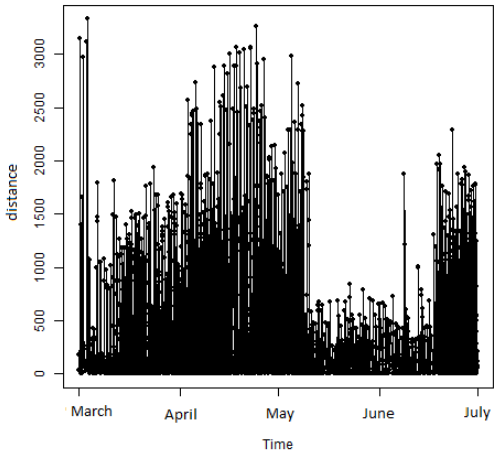


Figure 1.2: *Plot of the distance that the goose cover and the time.*
distance: the distance between two successive relocations, expressed in the units of the coordinates x,y (here in meters).

As we can see in Figure 1.2, the goose appears to move less than usual in June. The scientific advisor of the Society for the Protection of Prespa, George Katsadorakis, informs us that this is expected since greylag geese molt in June and so do not move considerably until they can fly again. The peak can be either due to false GPS location or something to disrupt its normal movement for example, to be chased by another animal.

1.2.1 The scatterplot

One way to study the home range of an animal is simply to examine the scatterplot of its location data. This is shown in Figure 1.3 for the greylag goose.

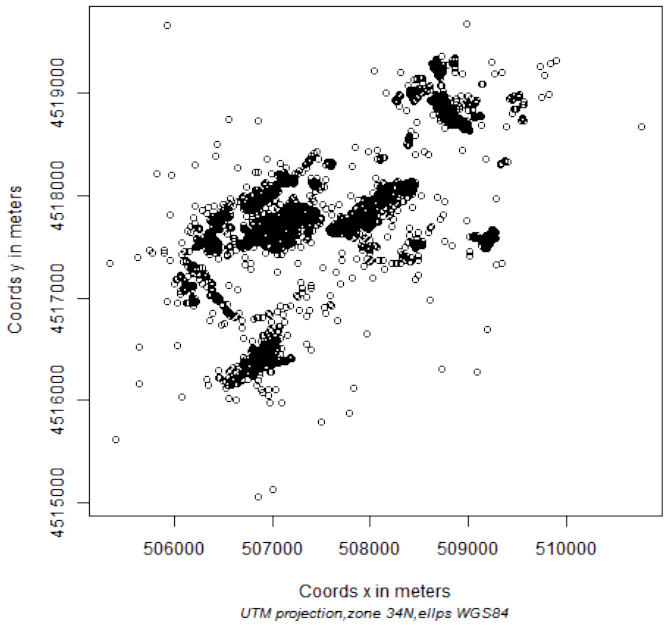


Figure 1.3: *The scatterplot reflects the location data collected on the greylag goose in Prespa, Greece.*

This scatterplot is quite useful for qualitatively understanding the goose's home range, but it has limitations, as it lacks information on the time order of the data, on the lengths of time intervals between time-consecutive data points and lacks quantitative information on the relative amounts of time the goose spends in various areas (Bullard 1991). In addition, as it is observed in our scatterplot, when we have large samples, it's too "busy" to convey information and any overlapping data points are presented as a single point. Thus, for example, the dense area located near the coordinates (507000,4516300) may be even denser than it appears.

The scatterplot for the greylag goose in google maps is shown in Figure 1.4:

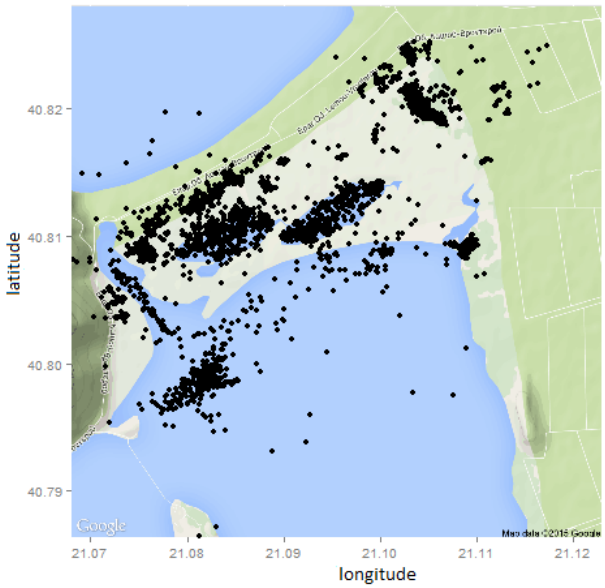


Figure 1.4: *The scatterplot in google maps. Longitude/Latitude projection.*

1.2.2 The connecting scatterplot

The lack of consideration given to the time order of the data may be partially rectified by drawing a *connecting scatteplot* of the data (Fig. 1.5). Unfortunately, this compounds a new problem related to the lengths of time intervals between time-consecutive data points. The straight lines may be a poor representation of the true path followed by the goose. Thus, the lines may falsely suggest the goose's presence at locations which in fact it avoided. Furthermore, the accuracy of each connecting line segment in the drawing is related to the goose's average speed over the corresponding time interval. When the goose's average speed is high, the line segment is more likely to be a good estimate of the goose's actual path than when the average speed is low. The connecting scatteplot does not reveal this information (Bullard 1991).

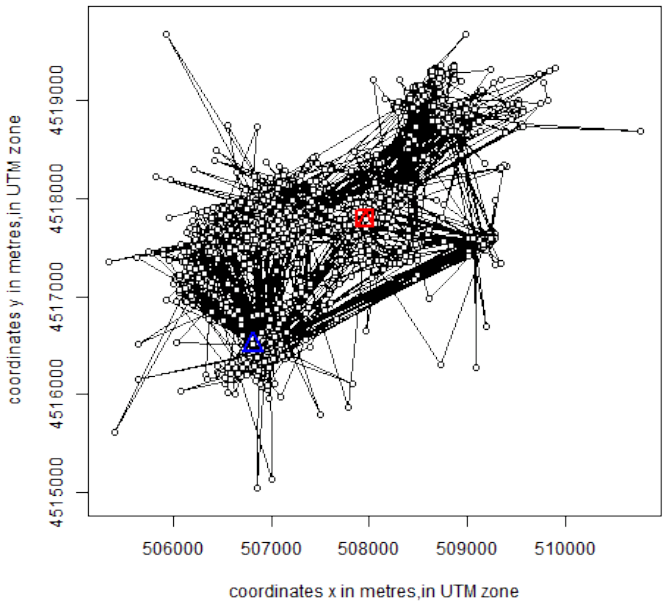


Figure 1.5: *The trajectory of the goose, monitored every 13-100 minutes from March to June. The initial and final relocations of the trajectory is indicated in blue and red, respectively.*

In order to eliminate all these problems, we need to estimate the probability density function, f, of the home range. This function is defined over the space of all points the goose might possibly visit and has the following properties:

- f is non-negative everywhere;
- The total volume beneath the surface defined by f is equal to 1;
- The volume beneath the surface defined by f and lying over a particular region R is equal to the probability that the goose is in R at random moment in time. This may also be thought of as the relative proportion of time that the goose spends in R.

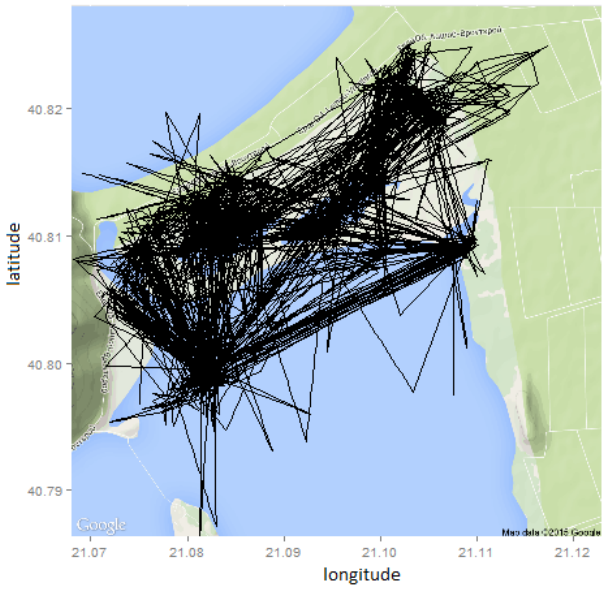


Figure 1.6: *The connecting scatterplot in google maps. Longitude/Latitude projection.*

Chapter 2

Kernel Density Estimation

The most acceptable method of home range analysis with *uncorrelated* locations is *Kernel Density Estimation* (KDE). Although this method has been used widely for GPS technology, it is prone to errors in proper bandwidth selection and violation of independence assumptions. The issue of autocorrelation or independence in location data has been dissected repeatedly by users of KDE for decades, and can be especially problematic with data collected with GPS technology as sampling frequency increases (Walter 2011).

2.1 Method

Intuitively, the kernel method consists of placing a kernel (a probability density) over each observation point in the sample. A regular rectangular grid is superimposed on the data, and an estimate of the density is obtained at each grid intersection, using information from the entire sample. The estimated density at each intersection is essentially the average of the densities of all the kernels that overlap that point. Observations that are close to a point of evaluation will contribute more to the estimate than ones that are far from it. Thus, the density estimate will be high in areas with many observations, and low in areas with few (Seaman & Powell 1996).

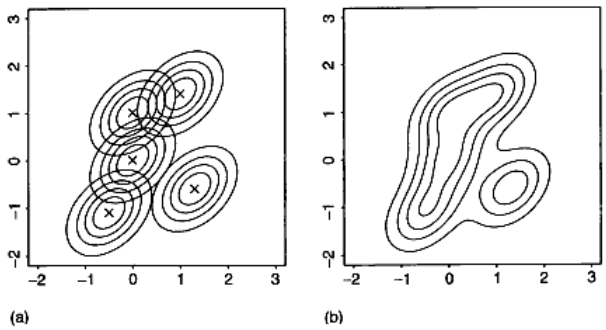


Figure 2.1: *Construction of a bivariate kernel density estimate: (a) kernel mass being centred about each observation (b) contour diagram of the resulting kernel estimate.*

The kernel density estimator for bivariate data of the UD is mathematically defined as:

$$\hat{f}(\mathbf{x};\boldsymbol{h}) = n^{-1}\sum_{i=1}^{n} K_{\boldsymbol{H}}(\mathbf{x}-\mathbf{X}_i),$$

where $\boldsymbol{H}$ is a symmetric positive definite 2x2 matrix called the *bandwidth matrix* depending on smoothing parameter(s) $\boldsymbol{h}$,

$$K_{\boldsymbol{H}}(\mathbf{x}) = |\boldsymbol{H}|^{-\frac{1}{2}} K(\boldsymbol{H}^{-\frac{1}{2}}\mathbf{x})$$

n is the number of data points (relocations), K is a bivariate kernel function, $\mathbf{x}$ is a vector of (x,y) coordinates describing the location where the function is being evaluated and $\mathbf{X}_i$ is a series of vectors whose coordinates describe the location of each observation i (the i^{th} relocation of the sample).

Several kernel functions can be used in the estimation process provided that:

$$\int_{R^2} K(\mathbf{x})d\mathbf{x} = 1 \text{ and } K(\mathbf{x}) > 0\ ,\ \forall \mathbf{x} \in \Re^2$$

where $\mathbf{x}$ is a vector containing the coordinates of a point on the plane.

Usually K will be a radially symmetric unimodal probability density function, such as the standard bivariate normal kernel density function (Fig. 2.2), which is defined as:

$$K(\mathbf{x}) = (2\pi)^{-1}\exp(-\tfrac{1}{2}\mathbf{x}^T\mathbf{x}).$$

Another possible kernel is the bivariate Epanechnikov kernel:

$$K_e(\mathbf{x}) = \begin{cases} \frac{2}{\pi}(1-\mathbf{x}^T\mathbf{x}), & \mathbf{x}^T\mathbf{x} < 1 \\ 0, & otherwise. \end{cases}$$

The choice of the kernel function has not as much influence on the results as the bandwidth selection, if the same bandwidth is used (Silverman 1986). Although the Epanechnikov kernel is slightly more efficient, the bivariate normal kernel is a common choice.

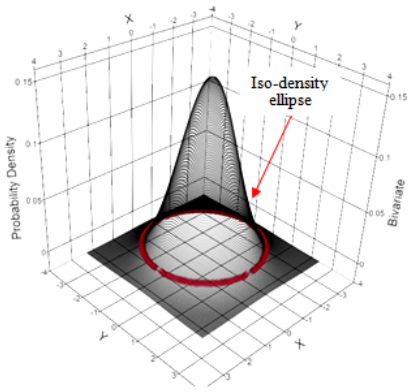


Figure 2.2: *Standard bivariate normal kernel density. The kernel is a probability density; the volume under the curve integrates to 1.*

2.1.1 Bandwidth

The bandwidth (smoothing parameter) h controls the "width" of the kernel functions placed over each point. The use of an identity matrix $\boldsymbol{H} \in S$, where $S = \{h^2 I : h > 0\}$ implies that the kernel placed on each data point is scaled equally in all directions. This restriction has the advantage that one only has to deal with a single smoothing parameter, but the considerable disadvantage that the amount of smoothing is the same in each coordinate direction. This may not be appropriate if for example the spread of the data points is very much greater in one of the coordinate directions than the others. At the next level, $\boldsymbol{H} \in D$, $\boldsymbol{H} = diag(h_1^2, h_2^2)$ one has the flexibility to smooth by different amount in each of the two coordinate directions. However, there are situations where one might wish to smooth in directions different to those of the coordinate axes. In this case the full bandwidth matrix , $\boldsymbol{H} \in F$, $\boldsymbol{H} = \begin{bmatrix} h_1^2 & h_2 \\ h_2 & h_3^2 \end{bmatrix}$ would be appropriate.

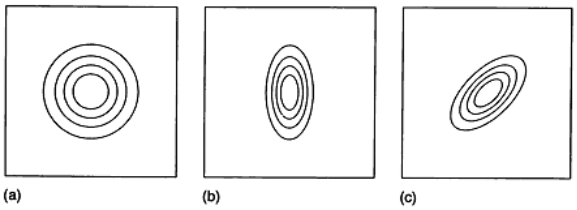


Figure 2.3: *Contour plots of kernels parametrised by (a) $\boldsymbol{H} \in S$, (b) $\boldsymbol{H} \in D$ and (c) $\boldsymbol{H} \in F$*

$\boldsymbol{H}$ is of practical relevance to compare the performance of the smaller bandwidth matrix classes to the full class F. Obviously there will be some loss of efficiency due to using fewer parameters, but of interest is just how large this loss can be. This depends on the particular density shape and orientation with respect to the coordinate axes. Broadly speaking one often doesn't lose very much by using a diagonal bandwidth matrix although, full matrices are necessary in some circumstances (Wand & Jones 1993).

Just as for many other multivariate statistical procedures, it is probably best to pre-scale the data to avoid extreme differences of spread in the various coordinate directions. If this done then there will generally be no need to consider more complicated forms of the kernel density estimate than the form that involves a single smoothing parameter. An attractive intuitive approach is suggested by Fukunaga (1972). This approach is equivalent to linearly transforming the data to have unit covariance matrix, often called *sphering* the data, applying the simple kernel estimator to the sphered data and then "backtransforming" to obtain the density estimator of the original data. In case of multivariate normal data, sphering is appropriate. However, there is no corresponding theoretical support for sphering for estimation of general density shapes (Silverman 1986; Wand & Jones 1995).

The study extent is then gridded with evaluation points in which different kernels are summarized to produce a utilization distribution across the area of interest. The resulting utilization distribution is therefore sensitive to the resolution of the evaluation grid, and more importantly, to the bandwidth selection (i.e., smoothing parameter) of the kernels (Walter 2011).

2.2 Bandwidth Selection

The practical implementation of the kernel density estimator requires the specification of the bandwidth matrix. This choice is very important, as will lead to considerably different results in our estimation. In general, the problem of selecting a bandwidth matrix from the data for multivariate kernel estimation has received less attention in the literature than its univariate counterpart. However, many of the ideas for selecting the bandwidth of the kernel estimator that exist in the univariate case can be extended to a multivariate setting. In this section, we will discuss three of these ideas, in order to estimate the home range of the greylag goose.

2.2.1 $H_{reference}$

It is generally accepted in the kernel literature that the choice of the kernel K is not as important as the choice of the bandwidth. Thus, for a given kernel and sample size we can find the theoretical optimum choice of bandwidth by finding the value of h that minimizes the Mean Integrated Square Error (MISE). The MISE is defined as:

$$MISE(h) = E\int(\hat{f} - f)^2$$

where E denotes the expectation with respect to the random observations. For our bivariate case, the integration is over the plane.

The optimal bandwidth has been determined analytically for standard multivariate normal distributions. We will refer to this as the *reference bandwidth* (h_{ref}) after Worton (1995). For any number of dimensions of data being analyzed, the bandwidth for each dimension i ($i = 1, ..., d$) is defined as: $h_i = A\sigma_i n^{-\frac{1}{4+d}}$, where A is a constant that tailors the bandwidth to the particular kernel being used, d is the number of dimensions and σ_i is an estimate of the standard deviation of the data in dimension i (Silverman 1986).

Using a normal kernel, it can be shown that for the bivariate normal distribution with variance-covariance matrix $\begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}$, an estimate of the optimum h for large sample size n is:

$$\hat{h}_{ref} = \hat{\sigma} n^{-\frac{1}{6}},$$

where $\hat{\sigma} = \left\{ \frac{1}{2} \left[\hat{\sigma_x}^2 + \hat{\sigma_y}^2 \right] \right\}^{\frac{1}{2}}$ and $\hat{\sigma_x}^2$,$\hat{\sigma_y}^2$ are the estimated variances of the x and y coordinates of the relocations respectively. If these variances differ greatly it may worth rescaling the data so that the variances are equal before applying a kernel method (Worton 1989). Thus, the bandwidth matrix $\boldsymbol{H} \in S$, where $S = \left\{ \hat{h}^2_{ref} I_2 : h > 0 \right\}$. Also if an Epanechnikov kernel is used, h_{ref} is multiplied by 1.77 (Silverman 1986).

2.2.1.1 Discussion

Animal utilization distributions are seldom close to standard bivariate normal; they frequently have multiple modes (centers of activity) with differing heights and widths. Such distributions violate the assumption of normality and result in the choice of too large a bandwidth if the reference bandwidth is chosen. This is because the reference bandwidth treats the distribution as if it were a single unimodal normal and creates an estimate with the amount of smoothing that would be appropriate for such a distribution. Nonetheless, this bandwidth presents a plausible initial choice (Seaman & Powell 1996).

So estimation with h_{ref} typically is not reliable for use on multimodal datasets because it results in over-smoothing of home ranges and multimodal distribution of locations is typical for most species (Worton 1995; Seaman 1999). An important point to consider with previous investigations on bandwidth selection is that analyses used simulated data on only 10 - 1,000 locations for assessing reliability of h_{ref} (Seaman 1999; Lichti & Swihart 2011). Still, results from simulated datasets and real-world examples concluded that h_{ref} should not be used on multimodal data typical for most mobile species (Worton 1995; Seaman & Powell 1996; Hemson 2005).

2.2.1.2 Results

Kernel estimators were run on actual location data from a Bluethooth/GPS neckband transmitter of a greylang goose. The goose is tracked from March through June 2014 (sample size: $n = 4852$). First, consider kernel estimation of the UD density with h_{ref} bandwidth.

Using h_{ref} and the bivariate normal kernel, we found $\hat{h}_{ref} = 187.0615$, so
$\boldsymbol{H}=\begin{bmatrix} (187.0615)^2 & 0 \\ 0 & (187.0615)^2 \end{bmatrix}$.

Very similar estimates indeed were obtained when the above analysis was repeated using the Epanechnikov kernel and h_{ref} bandwidth (Fig. 2.4). This supports the theoritical findings that the precise form of the kernel used is unimportant as long as the bandwidth is appropriately selected. So for simplicity we used the bivariate normal kernel.

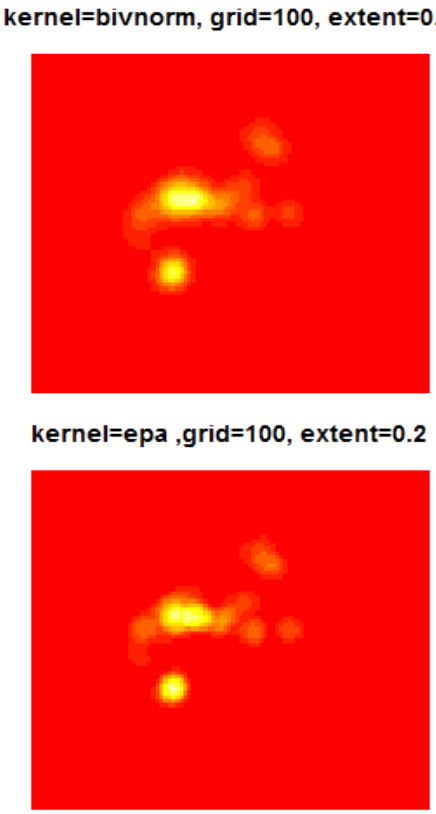


Figure 2.4: *Comparison of the Epanechnikov ($h_{ref} = 331.0988$) and the Bivariate Normal kernel ($h_{ref} = 187.0615$).*

The resulting home range size areas for several probability levels are given below (the units of the output areas are in km^2):

50%	55%	60%	65%	70%
1.293387	1.539189	1.808401	2.118580	2.469726

75%	80%	85%	90%	95%
2.885248	3.376852	4.008915	4.851665	6.226986

Once the utilization distribution has been estimated, the density is converted into a home range estimate. Contours connecting areas of equal density can describe any usage area of the home range.

The home range in raster mode for several probability levels is illustrated in Figure 2.5. The UD has modified so that, the value of a pixel is equal to the percentage of the smallest home range containing this pixel.

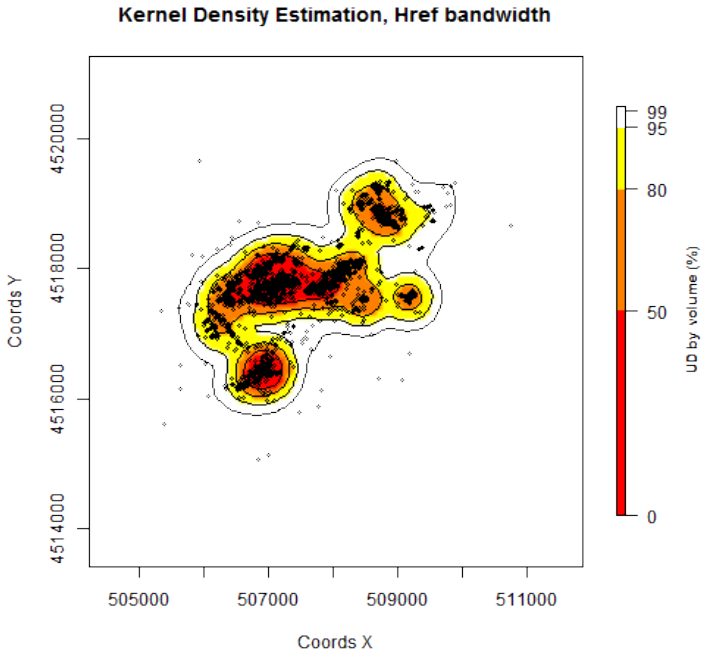


Figure 2.5: *Estimated home range of the goose using KDE h_{ref}. Contours represent 50, 80, 95 and 99% of the volume of the home range estimate; data points mark the GPS locations (extent=0.2,grid=100).*

The home range deduced from the UD as the minimum area on which the probability to relocate the animal is equal to a specified value 0.95 (95%) is shown in Figure 2.6. Including only, say, the smallest area in which the animal spent 95 percent of its time could exclude "occasional sallies" or areas the animal will never visit again.

Figure 2.6: *An 95% estimate of the goose home range derived from kernel density estimation with h_{ref} bandwidth (area=6.226986 km^2).*

2.2.2 Least Squares Cross-Validation

Another method for choosing the bandwidth is *least squares cross-validation* (LSCV). This process examines various bandwidths, and selects the one that gives a minimum score for the estimated error $M(h)$ (the difference between the unknown true density function and the kernel density estimate). If a standard bivariate normal density kernel K is used, the value of h is chosen to minimize:

$$M(h) = \frac{1}{n^2h^2}\sum_{i=1}^{n}\sum_{j=1}^{n} K^{\star}\left[\frac{X_i - X_j}{h}\right] + \frac{2K(0)}{nh^2},$$

where $K^{\star} = K^{(2)} - 2K$ and $K^{(2)}$ is a bivariate normal density with variance - covariance matrix $\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$.

The intuitive reason this method provides a good estimate is that:

$$E\left[M(h)\right] + \int f^2 \approx E\int (\hat{f}_h - f)^2.$$

Thus, by minimizing $M(h)$ we would hope to minimize $MISE(h)$ (Worton 1989).

Since the variances in the two dimensions may be unequal, bandwidths are selected by the following procedure. The data are standardized by dividing each coordinate by the standard deviation of the observations for that dimensions (Silverman 1986; Fukunaga 1972). Cross validation is performed on the standardized

data, which allows the program to select a single best bandwidth for the dataset. Then two bandwidths are created one for each dimension by multiplying the selected bandwidth by the standard deviation of each dimension of the data. This allows the amount of smoothing in each dimension to respond to the amount of variation in that dimension, effectively creating an asymmetrically elongated kernel when the data are distributed in an elongated distribution along the x and y axis. However, the kernel does not respond to diagonal elongation that results from covariance between the x and y coordinates (Seaman & Powell 1996).

2.2.2.1 Discussion

Both least squares cross-validation (h_{lscv}) and bias crossed validation (h_{bcv}) have been suggested instead of h_{ref} in attempts to prevent over-smoothing of KDE (Rodgers & Kie 2010). However, h_{lscv} and h_{bcv} have been minimally evaluated on GPS datasets because previous literature only evaluated datasets collected on VHF sampling protocols or simulated data that included at most 1,000 locations and did not represent actual animal distributions (Worton 1995; Gitzen 2006; Lichti & Swihart 2011). Least-squares cross validation, suggested as the most reliable bandwidth for KDE (Worton 1989), was considered better than plug-in bandwidth selection, discussed in Section 2.2.3, at identifying distributions with tight clumps of points but risk of failure increases with h_{lscv} when a distribution has a "very tight cluster of points" (Gitzen 2006; Pellerin 2008; Walter 2011).

2.2.2.2 Results

Using the method LSCV the score function $M(h)$ gave no minimum (Fig. 2.7). So we aren't able to find h_{lscv}.

This was almost expected as we know that LSCV method used for the parameter estimation is sensitive to large samples (sample *sizes* > 1000 locations) (Hemson 2005) and here $n = 4852$. Also, greylag goose's locations were truncated to a grid with multiple overlapping points so could be classified as having "very tight cluster of points".

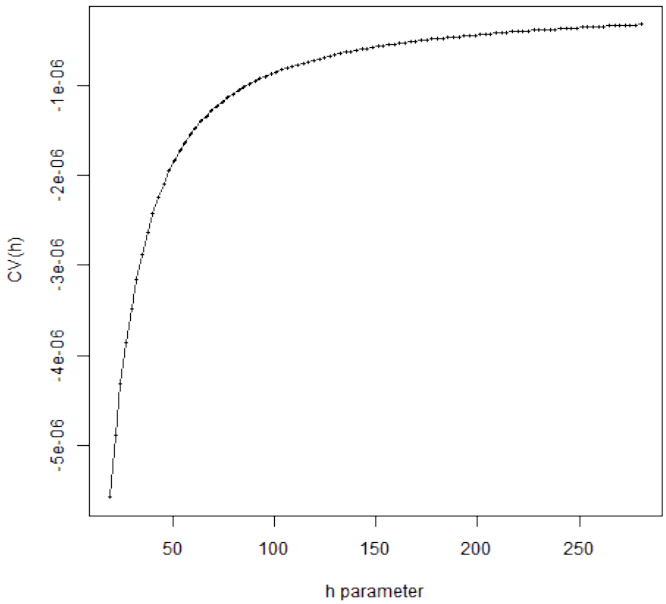


Figure 2.7: *The results of the LSCV minimization.*

One way to address lack of convergence due to large datasets is subsampling (Avery 2011) that can be used for crude estimates of home range using KDE with h_{lscv}. But subsampling can potentially remove important movement parameters or habitats used and will not result in the same estimate of home range size as the complete GPS dataset (Blundell 2001; Pellerin 2008; Rodgers & Kie 2010).

2.2.3 Plug-in bandwidth selection

Most first generation methods of bandwidth selection for density estimation (i.e., h_{lscv}) were developed before 1990 but advances in theory and technological capabilities has opened the door for second generation methods (Jones 1996). Second generation methods, such as the smoothed bootstrap and plug-in methods (often combined into the solve-the-equation plug-in method; Jones 1996), appear to be an improved alternative because of better convergence and reasonable trade-offs between bias and variance compared to first generation methods (Jones 1996; Duong & Hazelton 2003).

The plug-in method is based on the estimation of the Asymptotic Mean Integrated Square Error (AMISE), by replacing Ψ by estimator $\hat{\Psi}$:

$$\hat{AMISE}\,\hat{f}(\cdot;\boldsymbol{H}) = PI(\boldsymbol{H}) = n^{-1}(4\pi)^{-1}\,|\boldsymbol{H}|^{-\frac{1}{2}} + \tfrac{1}{4}(vech^T\boldsymbol{H})\hat{\Psi}(vech\boldsymbol{H}),$$

where Ψ is a matrix of functionals that depend on f and $vech\boldsymbol{H}$ is the vector of lower triangular half of H i.e., if H is diagonal matrix, $vechH = \left[h_1^2, 0, h_2^2\right]$ (Duong & Hazelton 2003).

The optimal bandwidth is :

$$\hat{H}_{PI} = \underset{\boldsymbol{H}\in F}{\operatorname{argmin}} PI(\boldsymbol{H})$$

2.2.3.1 Discussion

Debate about the appropriateness of second generation methods still exists with some authors claiming the estimates obtained with bivariate plug-in bandwidth selection perform poorly compared to first-generation methods (Loader 1999) while others showed it performed well even when analyzing dependent data (Hall 1995).

2.2.3.2 Results

For kernel estimation with the bandwidth plug-in we used a diagonal matrix since, "one often doesn't lose very much by using a diagonal bandwidth matrix" (Wand & Jones 1993).

We found $\hat{H}_{PI} = \begin{bmatrix} 3441.018 & 0 \\ 0 & 2502.519 \end{bmatrix}$.

The resulting home range size areas for several probability levels are given below (the units of the output areas are in km^2):

50%	55%	60%	65%	70%
0.41	0.50	0.61	0.74	0.86

75%	80%	85%	90%	95%
1.07	1.32	1.63	2.09	2.93

The home range in raster mode for several probability levels is illustrated in Figure 2.8. The bumps in H_{PI} KDE, are due to a point in an area with a low concentration of points.

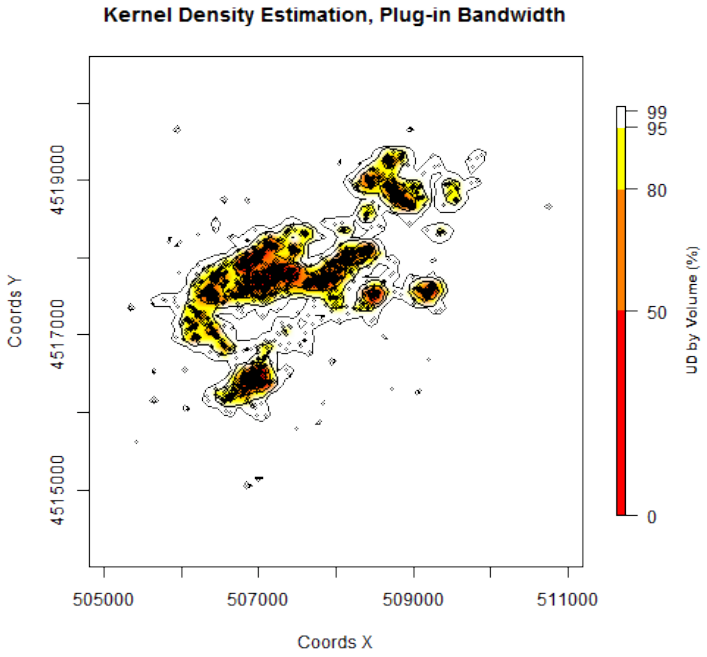


Figure 2.8: *Estimated home range of the goose using KDE H_{PI}. Contours represent 50, 80, 95 and 99% of the volume of the home range estimate; data points mark the GPS locations (extent=0.2,grid=100).*

The home range polygons generated with H_{PI} appear fragmented; they may be appropriate when studying a species in highly fragmented landscapes such as urban areas (Walter 2011).

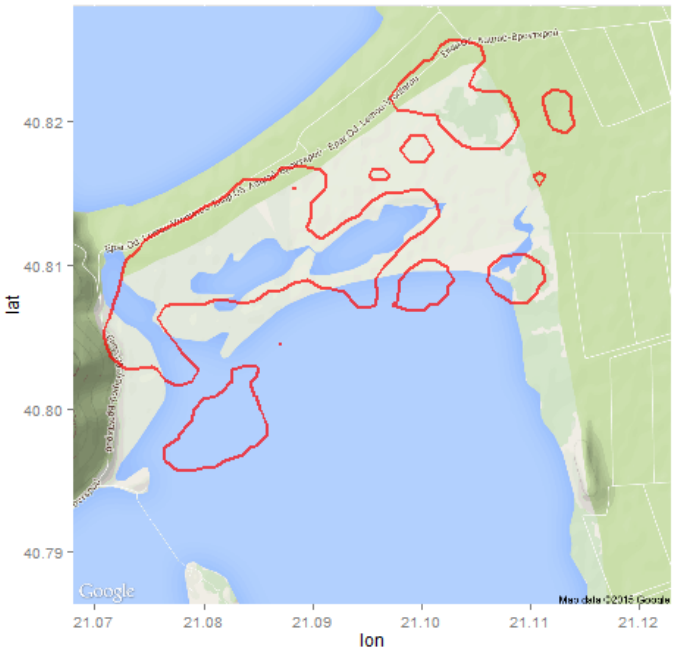


Figure 2.9: *An 95% estimate of the goose home range derived from kernel density estimation with H_{PI} bandwidth (area=2.93 km^2).*

Chapter 3

Brownian Bridge movement method

Global positioning system (GPS) telemetry is increasingly being used to study animal movements because it provides researchers the opportunity to almost continuously follow the movements of individuals for extended periods of time and over great distances. Using these types of data, Bullard (1999) and Horne (2007) describe a new model for estimating animal movements based on Brownian bridges.

A Brownian bridge is a continuous-time stochastic model of movement in which the probability of being in area is conditioned on starting and ending locations, the elapsed time between those points, and the mobility or speed of movement (Horne 2007).

An animal's movements define a path (i.e., trajectory), through an area, during a specified period of time from $t = 0$ to T_{total}. Let's assume that continuous observation of the animal is impossible but n discrete locations along the trajectory are available. Our interest is in modeling an animal's utilization distribution (i.e., the relative frequency of use of a two-dimensional area $A \in \Re^2$) during the period of observation $[0, T_{total}]$. Absent any a priori knowledge of movement patterns, it is natural to model such movement as a random walk or its continuous counterpart, Brownian motion (Fig. 3.1). An animal's frequency of use in an area is estimated by treating each of the n locations along the trajectory as known or approximately known, and using the properties of conditional random walk to model the expected movement path between each successive pair of locations. When a Brownian motion is extended for this situation (i.e., conditioned on the beginning and ending locations of each pair), the corresponding stochastic process is called a *Brownian bridge* (Horne 2007; Ross 1983).

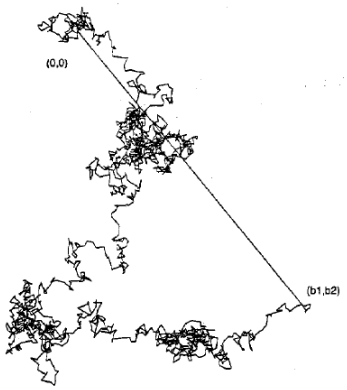


Figure 3.1: *Brownian motion from* $\overrightarrow{0} = (x_0, y_0)$ *to* $\overrightarrow{b} = (x_{total}, y_{total})$. *A Brownian motion may be thought of as a two dimensional random walk with infinitesimally small steps.*

Thus, the Brownian Bridge kernel method takes into account not only the position of the relocation, but also the path traveled by the animal between successive relocations (Fig. 3.2).

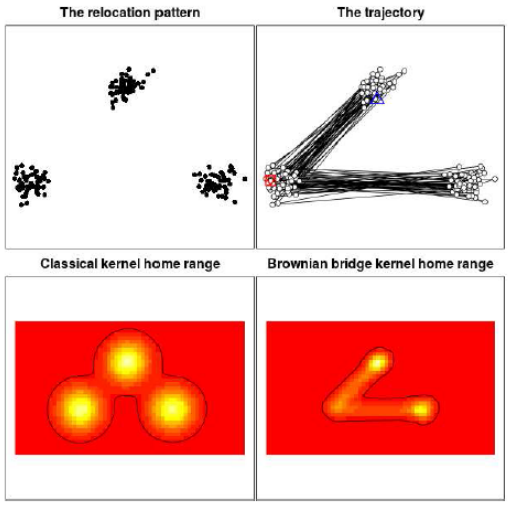


Figure 3.2: *The relocations are located into three patches. However, the order of the patches is not random. The BBMM allows us to identify that some areas between the patches are actually not used by the animal (Callenge, 2011).*

3.1 Method

Consider an animal that begins at a point $a = Z_0 = (x_0, y_0)$ at time $t = 0$ and is known to end at the point $b = Z_{total} = (x_T, y_T)$ at time $t = T$.

At time t, $0 < t < T$, the position of the animal will be unknown, but the probability density function describing its position is known to be a bivariate normal, $Z_t \sim N(\mu(t), \sigma_t^2 I_2)$, where

$$\mu(t) = a + \frac{t}{T}(b-a) \text{ and } \sigma^2(t) = \frac{t(T-t)}{T}\sigma_m^2.$$

Here, σ_m^2 is the variance of the *Brownian motion*, proportional to the speed of the animal, while $\sigma^2(t)$ is the variance of the animal's position as a function of time between known observations, taking the first observation to be at time $t = 0$ and the second to be at time $t = T$. I_2 is the 2x2 identity matrix.

The mean of this normal distribution moves from a to b proportional to the time between a and b $\left[i.e., \mu(t) = a + \frac{t}{T}(b-a)\right]$, and the variance equals 0 when $t = 0$, increases up to the midpoint in time between a and b, and then decreases back down to 0 when $t = T$ $\left[i.e., \sigma^2(t) = \frac{t(T-t)}{T}\sigma_m^2\right]$.

The probability density of a Brownian bridge with starting location a and ending location b, at any point z in time from $t = 0$ to $t = T$, is:

$$\hat{p_t}^{a,b,T}(a,z) = \phi(z; \mu(t), \sigma^2(t)),$$

where

$$\phi(z; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp[\frac{-(z-\mu)^2}{2\sigma^2}].$$

In addition, in tracking animal movements, *location error* is a prominent feature of most techniques for obtaining locations since the location is not known with absolute certainty. For example, triangulation techniques using radio tags yield some inaccuracy as does visual observation. Thus, the assumption of uncertain observations may not be unreasonable in estimating an animal's home range. Even in instances where absolute certainty is possible in locating the animal, the assumption of uncertainty is still reasonable (Bullard 1991).

Therefore, consider a Brownian Bridge that incorporates uncertainty in the starting and ending locations. To take this into account, Bullard (1991) let the starting and ending points to be random, with probability density functions $f_a(x)$ and $f_b(y)$, respectively, where x and y are position variables (two dimensional vectors) in $\Re^2$.

Thus, the probability of finding the animal in region A at time $t \in [0, T]$ is:

$$P(Z_t^T \in A) = \int\int P(Z_t^{x,y,T} \in A) f_a(x) f_b(y) dxdy =$$
$$= \int\int \left[\int_A \hat{p_t}^{x,y,T}(x,z)dz\right] f_a(x) f_b(y) dxdy.$$

To this point a Brownian bridge model has been described that estimates the probability of the animal being in an area A at a specific time t in the interval $[0, T]$. However, our main objective of study involves the frequency of use of an area over the entire time of observation.

First define the indicator function $1_A(x)$ that takes a value of 1 if x is in the region A and 0 otherwise. The random quantity:

$$\int_0^T 1_A(Z_t^T)dt$$

known as the *occupation time* for the region A, gives the amount of time during the observation period that the animal spends in A. Dividing by T and taking the expected value (E), we get the expected fraction of time in A. As a function of the region A, this yields a probability measure.

Our objective is to find the corresponding probability density function $h(z)$ such that:

$$E\left[\frac{1}{T}\int_0^T 1_A(Z_t^T)dt\right] = \int_A h(z)dz.$$

Indeed,

$$E\left[\frac{1}{T}\int_0^T 1_A(Z_t^T)dt\right] = \frac{1}{T}\int_0^T P(Z_t^T \in A)dt =$$
$$= \frac{1}{T}\int_0^T \left[\int\int\int_A \hat{p}_t{}^{x,y,T}(x,z)f_a(x)f_b(y)dzdxdy\right]dt =$$
$$= \int_A \left[\frac{1}{T}\int_0^T \int\int \hat{p}_t{}^{x,y,T}(x,z)f_a(x)f_b(y)dxdydt\right]dz.$$

Thus, the desired density function is given by:

$$h(z) = \frac{1}{T}\int_0^T \int\int \hat{p}_t{}^{x,y,T}(x,z)f_a(x)f_b(y)dxdydt.$$

This equation depends on the density functions f_a and f_b of the initial and final positions of the Brownian bridge, as well as the variance σ_m^2 of the underlying Brownian motion.

When the distribution of location errors, f_a and f_b, corresponds to *circular normal* distributions $N(a,\delta_a^2 I)$ and $N(b,\delta_b^2 I)$, respectively, the density function simplifies to:

$$h(z) = \frac{1}{T}\int_0^T \phi(z;\mu(t),\sigma^2(t))dt$$

where,

$\sigma^2(t) = Ta(1-a)\sigma_m^2 + (1-a)^2\delta_a^2 + a^2\delta_b^2$

and

$a = \frac{t}{T}$.

To avoid confusion of variance terms, note that the *variance of location error* is symbolized by δ^2. While the above expression does not have an explicit form, it can be approximated by discretizing time into arbitrarily small intervals of dt.

Model for *n* locations (*n*>2)

Consider the situation in which an animal's movements are monitored over an extended period of time, resulting in a series of space-time observations (Z_0,t_0) , (Z_1,t_1),..., (Z_n,t_n) collected during $T_{total} = t_n - t_0$.

Here the location errors are assumed to be normally distributed. Thus, the actual position of the animal at time t is modeled as a normal random variable $\boldsymbol{Z}_i \sim N(z_i, \delta_i^2 I_2)$, where $\boldsymbol{Z}_i$ is the i^{th} observed location and t_i is the corresponding time of that observation.

Given the n observations during the time interval $[0, T_{total}]$, and accounting for location error as described, the density function for the fraction of time at z during $[0, T_{total}]$ is:

$$\boxed{h(z) = \frac{1}{T_{total}} \sum_{i=0}^{n-1} \left\{ \int_0^{T_i} \phi(z; \mu_i(t), \sigma_i^2(t)) dt \right\}}$$

where

$T_i = t_{i+1} - t_i$,
$\mu_i(t) = z_i + a_i(z_{i+1} - z_i)$,
$\sigma_i^2(t) = T a_i(1-a_i)\sigma_m^2 + (1-a_i)^2\delta_i^2 + a_i\delta_{i+1}^2$
$a_i = (t - t_i)(T_i)^{-1}$,

δ_i is the standard deviation of the location error corresponding to the i observation, $i = 1, 2, ..., n$, σ_m^2 is the variance of the underlying Brownian motion and σ_i^2 is the variance of the animal's position as a function of time between known observations.

Because the BBMM estimates the probability that the animal occurred in an area over the analysis period, there is a direct application for estimating animal home range (Bullard 1999; Powell 2000; Horne 2007).

3.1.1 Parameter Estimation

The BBMM is dependent on *time-specific location data* (geographic position (x, y) and time stamps (t) of the location), *the distribution of location errors* and the *Brownian motion variance parameter* σ_m^2.

The time interval between locations is a factor that can be manipulated by the researcher. By decreasing the amount of time between successive locations, the uncertainty of the actual path can be reduced. As the time interval increases, there is less and less certainty of the actual path and this uncertainty is reflected in a flatter probability distribution between observed locations.

The location error is assumed to be normally distributed, with mean centered on the estimated location and variance either known (reported by the manufacturer of the GPS collar) or estimated via independent experiment. As technological advances continue to shrink the time between locations, the location error becomes the ultimate limit on the accuracy of estimating animal movements.

However, σ_m^2 which is related to animal's mobility, is a feature of the particular animal under observation. An empirical estimate of σ_m^2 can be obtained from the location data used to construct the BBMM by assuming that the path connecting any two observed location is a Brownian bridge.

To estimate σ_m^2 assume that n is even and consider the independent Brownian bridges on the nonoverlapping time intervals $[t_0,t_2]$,$[t_2,t_4]$,..., $[t_{n-2},t_n]$ while regarding the in-between observation times $t_1,t_3,...,t_{n-1}$ as independent observations from these Brownian bridges (Fig. 3.3). Under the assumptions of the Brownian bridge model, this yields a sample of $n/2$ independent odd observations, $Z_1,Z_3,...,Z_{n-1}$ that are normally distributed, $\boldsymbol{Z}_i \sim N(\mu_i(t_i),\sigma_i^2(t_i)I_2)$ where,

$\mu_i(t_i) = \boldsymbol{Z}_{i-1} + a_i(\boldsymbol{Z}_{i+1} - \boldsymbol{Z}_{i-1})$;

$\sigma_i^2(t) = T_i a_i(1-a_i)\sigma_m^2 + (1-a_i)^2\delta_{i-1}^2 + a_i\delta_{i+1}^2$;

$a_i = \frac{(t_i - t_{i-1})}{T_i}$;

and

$T_i = t_{i+1} - t_{i-1}$.

This allows us to construct the following likelihood function for odd locations:

$$L = \prod_{i=1}^{n-1} \frac{1}{2\pi\sigma_i^2(t_i)} \exp\left\{ \frac{-|\boldsymbol{Z}_i - \mu_i(t_i)||\boldsymbol{Z}_i - \mu_i(t_i)|^\top}{2\sigma_i^2(t_i)} \right\}. \quad (1)$$

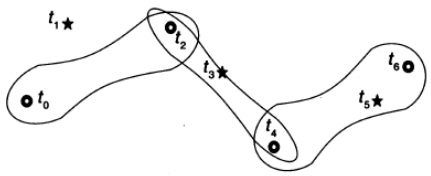


Figure 3.3: *Example of three Brownian bridges connecting even observations at time intervals* $[t_0,t_2]$,$[t_2,t_4]$ *and* $[t_4,t_6]$. *The in-between observations at times* t_1,t_3 *and* t_5 *are independent observations from these Brownian bridges and can be used to estimate the Brownian motion variance.*

Here $\top$ denotes transpose. So if the standard deviation of the location error (δ) for each location assumed to be known then, σ_m^2 is estimated by leave-one-out method (a minimum of three locations is required by the likelihood calculation) by numerically optimizing the likelihood function over values of σ_m^2. Thus σ_m^2 contains both information on how straight a movement path is, as well as how much a path varies in speed and the scale of movements (Horne 2007).

The interaction between the parameter σ_m^2 and δ^2 is shown in the following figure (Fig. 3.4). If the first combination is treated as standard, one notices that as σ_m^2 grows larger, the bridge widens, an expected result, since a faster animal covers more ground than a slower one and is therefore more likely to meander. So when σ_m^2 is small, the bridge becomes narrow, since a very slow animal must take a more direct path. Also as δ^2 grows larger, the bridge approximates a single bivariate normal distribution because the large uncertainty in the end positions "drown out" the part of the pdf dictated by the Brownian motion. Finally, when δ^2 is small, the bridge has pointed ends because of little uncertainty in the end positions (Bullard 1991).

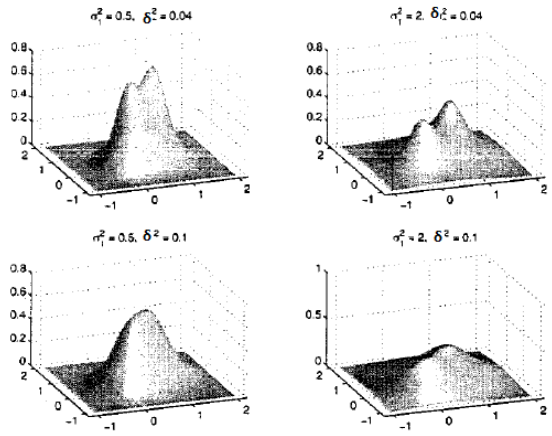


Figure 3.4: *Four graphs of a Brownian bridge from a=(0,0) to b=(1,1) in time* $t \in [0, 1]$.

3.2 Discussion

The BBMM is based on the properties of a conditional random walk between locations. Although it is certain that most animals do not move in a truly random fashion, animals usually stick to their favored haunts rather than randomly wander. In the absence of any other information on how an animal moved from one location to another, a Brownian bridge can serve as a useful approximation or null model of the actual movement process. However, violating the assumption of random movement between pairs of locations may become much more prominent as the time interval between locations increases. In this situation, animal movements between locations separated by long time intervals are more likely to reflect a biased random walk (i.e, toward the home range center) than a simple random walk between locations (Horne 2007). Benhamou & Cornelis (2010) developed *"Move-*

ment Based Kernel Estimation", an approach similar to Brownian bridge method that takes place in the framework of the biased random walk model. Here we weren't able to illustrate this method, because it needs a variable "activity" which defines if the animal is resting or not using a sensor detecting head movements of the animal, included in the GPS collar.

In addition, here a single estimate of the Brownian motion variance parameter is used for all pairs of locations. Because this parameter is related to the mobility of the animal, it would be reasonable to consider different variances for different behaviors. If researchers can a priori identify these periods, separate variance parameters could be estimated for each period. These different estimates could then be incorporated into the BBMM to more accurately depict animal movements (Horne 2007). Kranstauber (2012) developed a new method *"Dynamic Brownian Bridge Method"* that improves the BBMM by allowing for changes in behavior. We will see this approach in the next chapter.

3.3 Results

We assumed that the distribution of location error was circular normal, same standard deviation in x and y direction and there is no correlation between those two directions. Although the assumption of normally distributed errors is appropriate for GPS telemetry, this may not hold for locations collected using other satellite systems. Furthermore, we used a single estimate of the variance for all location errors to simplify calculations ($\hat{\delta}^2 = \hat{\delta}_1^2 = \hat{\delta}_2^2 = ... = \hat{\delta}_{4852}^2$). However, if researchers have reason to believe that each location has a unique error (Lewis 2007) this can be easily incorporated into the BBMM.

For the GPS used in the greylag goose study, the standard deviation of the location error reported by the manufacturer is $\hat{\delta} = 2.5$ meters.

The Brownian motion variance (σ_m^2) was estimated using the method of maximum likelihood described above. The optimum value is 10.1291 (Fig. 3.5).

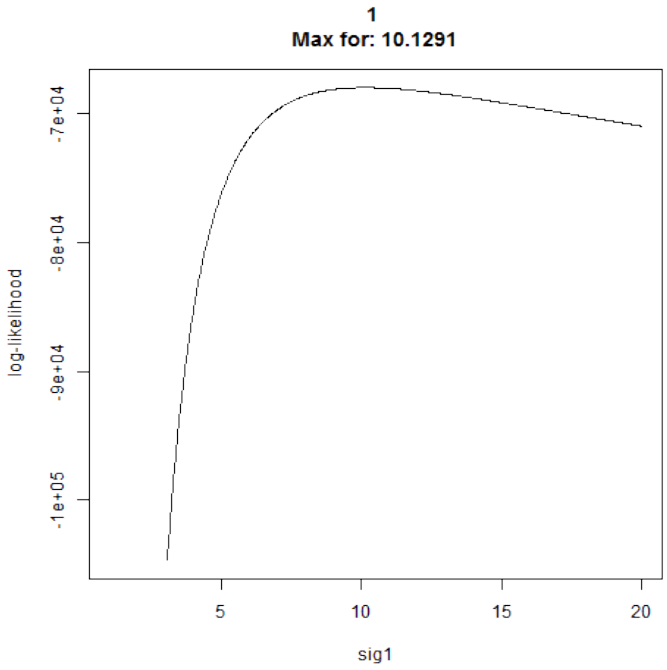


Figure 3.5: *The maximum likelihood estimation of* σ_m^2 *in the range* $[1,20]$*, using* $\hat{\delta} = 2.5$ *meters.*

The resulting home range size areas for several probability levels are given below (the units of the output areas are in km^2):

50%	55%	60%	65%	70%
1.452034	1.73604	2.079218	2.471827	2.912095

75%	80%	85%	90%	95%
3.473174	4.098021	4.880006	5.937952	7.762619

Using these parameters, an estimate of the utilization distribution was determined using the BBMM. The home range in raster mode for several probability levels is illustrated in Figure 3.6.

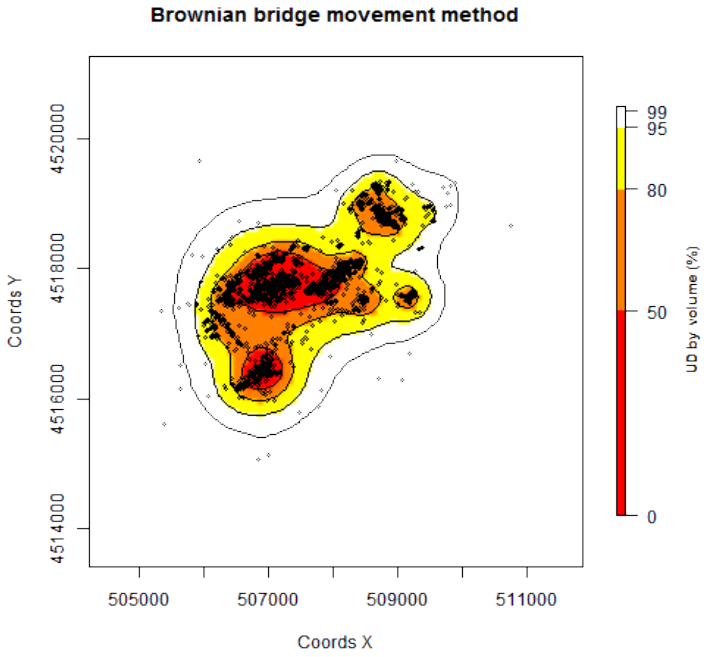


Figure 3.6: *Estimated home range of the goose using BBMM. Contours represent 50, 80, 95 and 99% of the volume of the home range estimate; data points mark the GPS locations (extent=0.2,grid=100).*

Figure 3.7: *An 95% estimate of the goose home range derived from BBMM (area=7.762619 km^2).*

In addition we examined two different approaches to estimate the telemetry error from the data itself if this, was unknown. The first approach was to use the variable "Estimated horizontal accuracy" (HACC) that is available in the data frame (see Figure 1.1). But for this approach a variable "Estimated vertical accuracy" (VACC) would be needed. Then due to the fact that the location error is circular normal (same standard deviation in x and y direction and there is no correlation between those two directions), we could estimate the standard deviation for both directions and average them. The second approach was to use the log-likelihood function (Eq.1) to estimate both σ_m^2 and δ with respect to both parameters, contrary to common practice, where δ is fixed. Figures 3.8 and 3.9 illustrate that the log-likelihood has a ridge shape but both parameters appear to be estimable. From Figure 3.9, the Maximum Likelihood Estimations of the parameters are in the range $\hat{\delta} = 20 - 22$ and $\hat{\sigma}_m^2 > 30$. Thus Maximum Likelihood Estimation of $\hat{\delta}$ is an order of magnitude higher than the reference value $\hat{\delta} = 2.5$ from the GPS manufacturer. Although there is no obvious reason not to use this approach, it needs further examination.

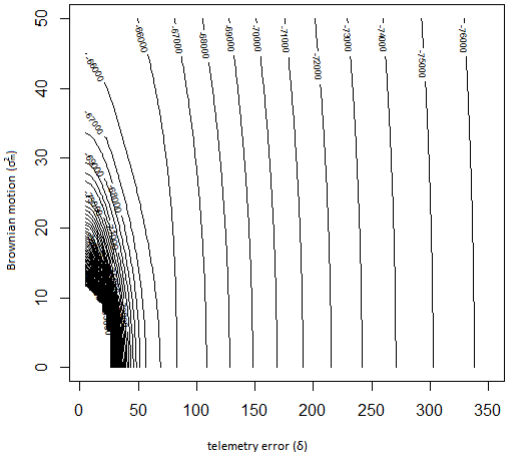


Figure 3.8: *Contours of the log-Likelihood function (Eq.1) calculated using the function* `"liker"` (`adehabitatHR package`). *Brownian motion ranging from 1 to 40, the telemetry error from 10 to 350* (`length.out=50`) *and the number of contours levels is 50.*

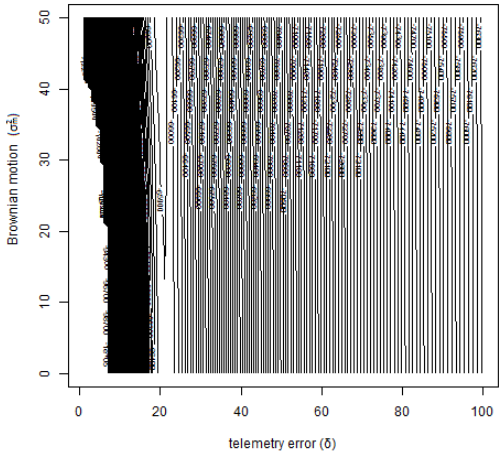


Figure 3.9: *Contours of the log-Likelihood function (Eq.1) calculated using the function* `"liker"` (`adehabitatHR package`). *Brownian motion ranging from 1 to 40, the telemetry error from 1 to 100* (`length.out=50`) *and the number of contours levels is 500.*

Chapter 4

Dynamic Brownian Bridge Movement Method

The recently developed Brownian bridge movement model (BBMM) has advantages over traditional methods because it quantifies the utilization distribution of an animal based on its movement path rather than individual points and accounts for temporal autocorrelation and high data volumes.

However, the BBMM can be improved as it does not take full advantage of the information contained in animal tracks. In particular, the BBMM assumes animal movement patterns within a track to follow one constant behavior as defined by the variance of the Brownian motion (σ_m^2), which quantifies how diffusive or irregular the path of an animal is. However, animal movement is actually composed of a succession of behaviourally distinct movement patterns (Morales 2004; Jonsen, Flemming & Myers 2005; Bailey 2008; Gurarie, Andrews & Laidre 2009). For example, within a day, animals may move in different ways when foraging versus travelling between sites, and almost all species break their day into periods of movement and rest (i.e., nocturnal, diurnal). On broader scales many species change their movement over the year or lifetime, for example migratory animals move over a small range when breeding but then make long distance movements for migration. Thus, estimating σ_m^2 for an entire trajectory will cause this parameter to be overestimated in some parts along the trajectory and underestimated in others. Overestimating σ_m^2 leads to an imprecision in the UD and thus wider UD areas; whereas underestimating σ_m^2 results in a false precision and too narrow UD areas (Kranstauber 2012).

Recently, Gurarie, Andrews & Laidre (2009) introduced the *Behavioural Change Point Analysis* (BCPA) to statistically determine where along an animal's trajectory changes in the behavioural state occur based on changes in the underlying movement patterns. The BCPA uses likelihood comparisons in a moving window to identify change points and quantifies the variation in the underlying movement

parameters along a trajectory. Kranstauber (2012) proposed a method that combines the BBMM with an approach similar to the BCPA to provide a dynamic and more accurate estimate of σ_m^2 along a path. This new movement analysis improves the estimation of UD, particularly for long complex animal journeys. In addition, adjusting σ_m^2 based on changes in movement patterns will provide insight into changes in behaviour along trajectories, very much like the original intention of the BCPA (Gurarie, Andrews & Laidre 2009).

4.1 Method

A Brownian bridge UD requires, in addition to the geographic position (x and y) and the timestamps (t) of the locations, the variance of the Brownian motion (σ_m^2) and a vector containing the standard deviations of the telemetry errors ($\boldsymbol{\delta}$). The error $\boldsymbol{\delta}$ can be derived empirically from field tests and is a property of the locations. The geographic positions together form the matrix $\boldsymbol{Z}$, where $\boldsymbol{Z}_i$ represents the x and y coordinates of location i and i ranges from 1 to n.

The variance of the Brownian motion σ_m^2 is a property of the intervals between locations, hereafter referred to as segments, and is estimated from the trajectory for a series of locations $\boldsymbol{Z}$ by maximizing the likelihood function using only odd values for i (Horne 2007):

$$L_1 = \prod_{i=1}^{n-1} \frac{1}{2\pi\sigma_i^2(t_i)} \exp\left\{ \frac{-|\boldsymbol{Z}_i - \mu_i(t_i)||\boldsymbol{Z}_i - \mu_i(t_i)|^\top}{2\sigma_i^2(t_i)} \right\}, \quad (1)$$

where

$\mu_i(t_i) = \boldsymbol{Z}_{i-1} + a_i(\boldsymbol{Z}_{i+1} - \boldsymbol{Z}_{i-1})$;
$\sigma_i^2(t_i) = T_i a_i(1-a_i)\sigma_m^2 + (1-a_i)^2\delta_{i-1}^2 + a_i\delta_{i+1}^2$;
$a_i = \frac{(t_i - t_{i-1})}{T_i}$;

and

$T_i = t_{i+1} - t_{i-1}$,

where δ_i is the standard deviation of the telemetry error corresponding to the i observation, $i = 1, 2, ..., n$.

The above equation assumes σ_m^2 to be the same along the entire path. However, Kraustauber (2012) suggested to use Eq.1 on subsections of trajectories to quantify a localized movement pattern of an animal and thus obtain a more refined UD.

The log-likelihood for a sliding window split in two parts changing at location b, the breakpoint, is calculated by:

$$\log(L_2(\sigma_m^2|Z_{1,w,b})) = \log(argmax_{\sigma_{m,1}^2 \in [0,\infty)}(L_1(\sigma_{m,1}^2|Z_{1,b}))) + \\ + \log(argmax_{\sigma_{m,2}^2 \in [0,\infty)}(L_1(\sigma_{m,2}^2|Z_{b,w}))) \quad (2)$$

where

$\boldsymbol{Z}_{i,j}$ is a subset of $\boldsymbol{Z}$, w is the size of sliding window, b is the location of the breakpoint within the sliding window and m is the margin size.

To estimate σ_m^2 for a subsection of a trajectory, Gurarie, Andrews & Laidre (2009) suggested to use an adjusted version of the BCPA that allows sudden as well as gradual changes in behavior. Within a sliding window with w locations, the log-likelihood of using just one value of σ_m^2 for the whole window (Eq.1) is compared to the log-likelihood of a window split in two parts (Eq.2) by comparing the Bayesian Information Criterion (BIC) values.

The procedure can summarized as follows:

1. Select a window of length w ($w \leq n$). Only odd values can be used because the likelihood estimation of σ_m^2 works on the basis of using every second location as an independent observation.

2. Select a margin of size m applied at start and end of each window in which no breakpoints could be estimated. A minimum of three locations is required.

3. Find the potential breakpoints b, $m \leq b \leq w - m$, again only odd values can be used.

4. Calculate the log-likelihood using Eq.2 at *each* breakpoint.

5. Calculate the log-likelihood using Eq.1 for just one σ_m^2 for the whole window w.

6. Obtain the Bayesian Information Criterion (BIC) value for each model.
$$BIC = -2n\log(L(\sigma_m^2|Z)) + d\log(n),$$
where d is the number of the parameters in each model.

7. The model with the lower BIC is preferred. Thus, we can see if there is a breakpoint or not.

8. Shift the window forward by one data point and repeat.

The sliding window will produce several estimates for each segment. These estimates will be averaged into one mean value per segment.

Also, due to the fact that not the same amount of σ_m^2 will be obtained at the beginning and the end of the track, the segments that don't have the maximum amount of estimates for σ_m^2 will be omitted. Valid estimates for σ_m^2 can be obtained only in the interval $[m, w-m]$ as there is the restriction that breakpoints can't occur in the margin of the sliding window.

Finally, after obtaining σ_m^2 for segments, the UD can be calculated according to Horne (2007). The difference it is that σ_m^2 varies, therefore is refered as *dynamic Brownian Bridge Movement Model* (dBBMM).

For example, suppose the window width is 11 locations and we select a margin of 3 locations (minimum=3), applied at the start and end point of each window in which no breakpoints can be estimated. The potential breakpoints will be 3, 5 and 7 (Fig. 4.1). We then calculate the BIC value for each model, $BIC_{w=3}$ (d=1), $BIC_{b1=3}$ (d=2), $BIC_{b2=5}$ (d=2) and $BIC_{b3=7}$ (d=2). The model with the lower value of BIC is the one preferred. Afterwards, the window will go forward by one data and the procedure will be repeated.

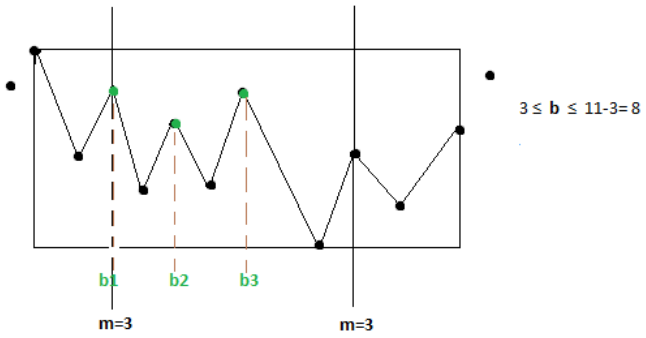


Figure 4.1: *Schematic description of σ_m^2 estimation in one window of length 11 and margin m=3 for dynamic Brownian bridges.*

Increasing the size of the sliding window (enlarging w) increases reliability in σ_m^2 estimation at the cost of missing short term changes in the variation parameter. In the other hand, increasing the margin size (m) enhances the power to identify "weak" breakpoints at the cost of not detecting breakpoints within the margin. The choice of m and w should be biologically informed and is determined by the time interval that changes in behaviour are expected to occur. However, for regularly sampled tracks, the condition $T_{change} > wT_{int}$ should be satisfied, where T_{change} is the smallest interval between expected behavioural changes and T_{int} the time between locations. This will ensure that every possible break can be described.

Window sizes larger than T_{change} could result in detecting either the onset or offset of a behaviour but not both.

It is important to note that optimal values for w and m are track specific and should not be generalized across projects and/or species. The main consideration should be the time scale of targeted behavioural changes.

4.2 Discussion

Dynamically estimating σ_m^2 for Brownian bridges provides two major advances. First, it improves on the estimation of the UD of the Brownian bridge movement models for behaviourally heterogeneous animal tracks by relaxing the assumption of a fixed σ_m^2. Second, the variation of σ_m^2 along a trajectory provides insight into variation in animal behaviour. Dynamic Brownian bridges method makes it possible to analyse entire tracks that include different behavioural types. In addition, dBBMM would also work for situations where the range of behaviour is unknown and therefore can not be identified by experts (Kraustuber 2012).

Kraustuber (2012) also showed that dBBMM performs better than, or at least as well as the traditional BBMM with a constant σ_m^2 . The performance of a dynamic estimation of σ_m^2 increased as the characteristics of the path before and after the breakpoint became increasingly dissimilar. This shows that the dBBMM and BBMM perform similar on tracks with low variation in movement pattern. It is important to highlight that the dBBMM produced better estimation of the home range particularly in cases where locations were randomly sampled, proving its power for nonregularly sampled tracks (e.g. missed GPS fix attempts).

4.3 Results

We applied the dBBMM to the track of the greylag goose. The trajectory was obtained using a GPS neckband transmitter that produced 4852 locations over a duration of 4 months (May to June 2014) in Prespa, Greece.

From the histogram in Figure 4.2 below we can see that the majority of the time lags between successive relocations is between 1 and 50 minutes. An approach as suggested by Benhamou (2011) would be to use the median of the time lags of the goose, which is $20.08333 \sim 20.08$ minutes. Thus, the GPS take approximately 3 fixes per hour, 72 fixes per day.

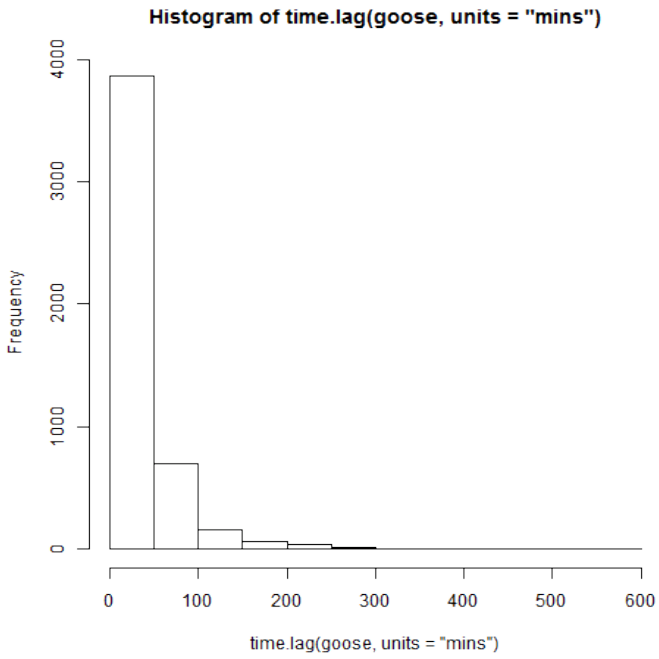


Figure 4.2: *Histogram of the time differences between successive relocations of the goose.*

Because no field measurement on the location error was available, it is reported by the manufacturer of the GPS collar to be 2.5 meters. We also assumed the same error, which is reasonable for GPS quality data (Frair 2010). Although we used one single location error along the track, there is no technical limitation to using differing location errors with the dynamic Brownian Bridge movement model as used by Lewis (2011) in combination with the Brownian Bridge movement model.

In order to detect diurnal changes in the behavior of the goose, we used a window size of 33 locations with margins of 13 locations, which translated into a window length of 11 hours ($T_{change} > wT_{int} = 33 * 20.08$ minutes in this case). It is important to note that regardless of the choice for the size of margins and window size, the dBBMM generally performs better than the BBMM approach (Kranstauber 2012).

To assess what a varying σ_m^2 could reveal about the behaviour of an individual, we plotted σ_m^2 over time (Fig. 4.3). The σ_m^2 estimates from the tracks showed that variation was low during the June, coinciding with the molting period that goose can't fly. These results highlight that a flexible σ_m^2 estimation can not only be used for calculating a UD, but can also indicate changes in the behavioural state of an individual.

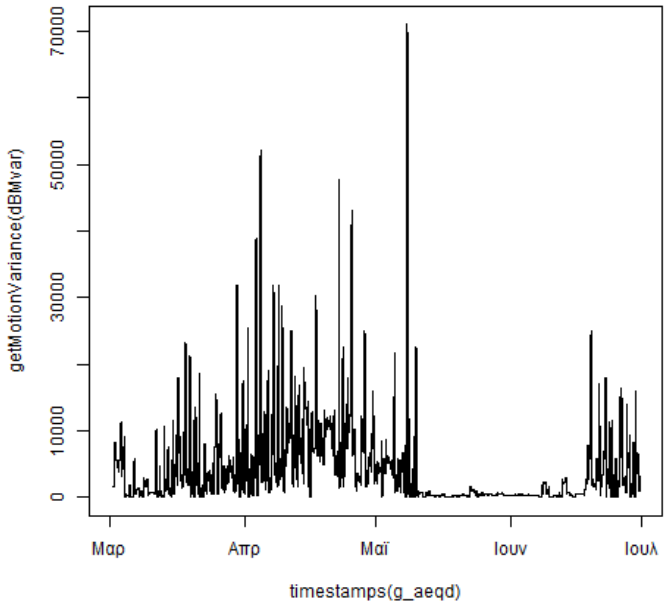


Figure 4.3: *The variation of σ_m^2 over time, where higher values indicate more irregular movement.*

The resulting home range size areas for several probability levels are given below (the units of the output areas are in km^2):

50%	55%	60%	65%	70%
0.6	0.78	1.02	1.33	1.72

75%	80%	85%	90%	95%
2.23	2.89	3.73	4.9	6.82

We then estimated the Utilization Distribution in raster mode, with the dynamic Brownian Bridge Method Movement within a raster grid size 100 and extent 0.2. Also, dBBMM code requires the coordinates to be in aeqd projection, which stands for Azimuthal Equidistance. This is different to the UTM projection we have used for the other methods but we weren't able to change it. The estimated home range in raster mode for several probability levels is illustrated in Figure 4.4.

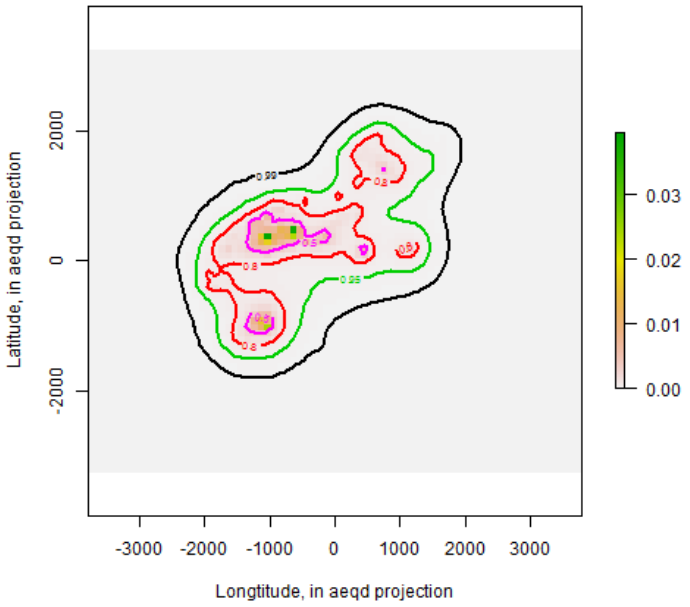
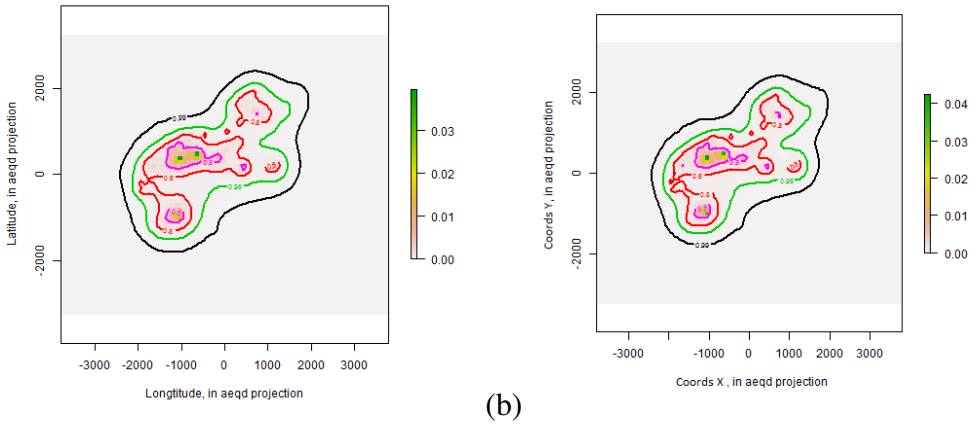


Figure 4.4: *Estimated home range of the goose using dBBMM (window=33,margin=13). Contours represent 50, 80, 95 and 99% of the volume of the home range estimate. The extent is 0.2 and the grid is 100.*

Figure 4.5: *An 95% estimate of the goose home range derived from dBBMM with a window size of 35 locations and margins of 15 locations (area=6.82 km^2).*

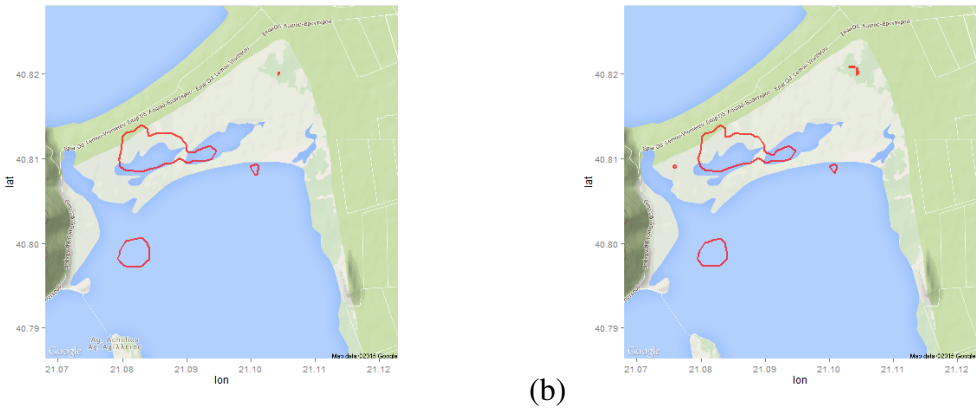
An alternative approach would be to use the mean of the time lags of the goose, which is 36.18822 $\sim$ 36.18 minutes. Thus, the GPS takes approximately 1.6 fixes per hour. In order to detect diurnal changes in the behavior of the goose, we used a window size of 23 locations with margins of 9 locations and recalculated the utilization distribution. The result is shown in Figure 4.6.



(a) (b)

Figure 4.6: *The home range of the goose using dBBMM with (a) a window size of 33 locations and margins of 13 locations and (b) a window size of 23 locations and margins of 9 locations. In both cases the extent is 0.2 and the grid is 100.*

A comparison of 50% estimates of the goose home range derived from the dBBMM with different window sizes and margins is shown in Figure 4.7.



(a) (b)

Figure 4.7: *A comparison of 50% estimates of the goose home range derived from the dBBMM with (a) a window size of 33 locations and margins of 13 locations and (b) a window size of 23 locations and margins of 9 locations.*

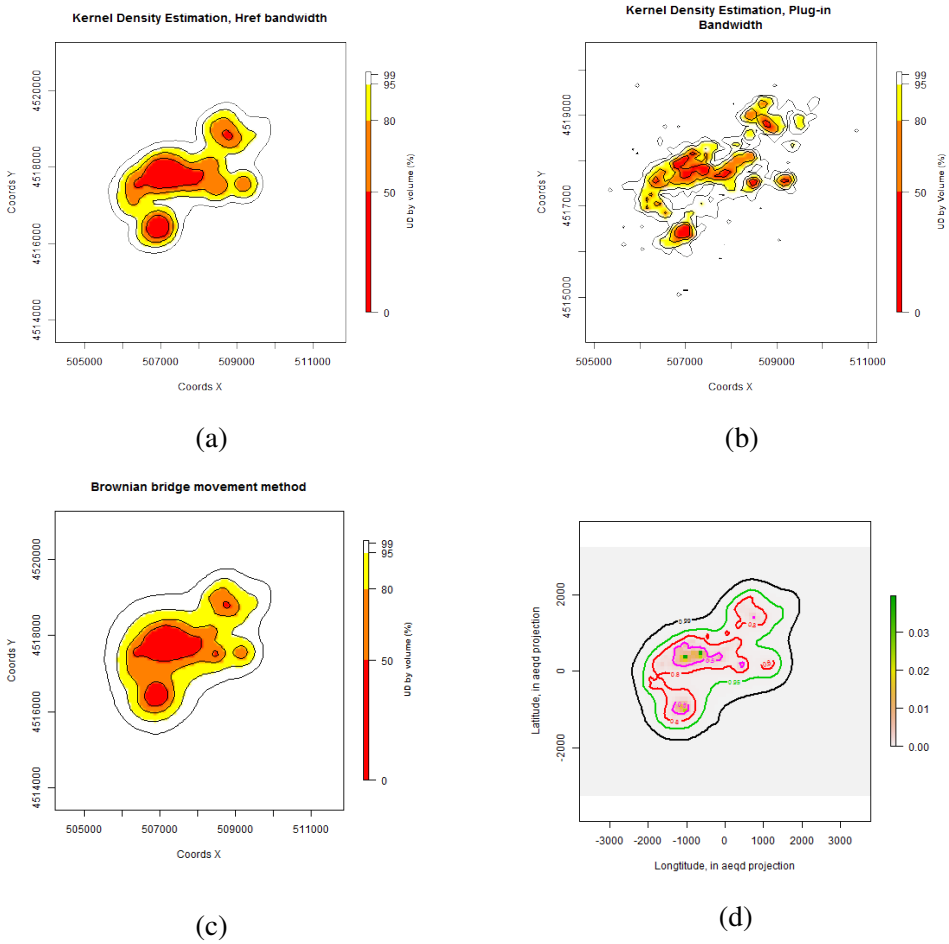
Chapter 5

Comparison of home range estimators

5.1 Comparison of aggregate data

For comparison purposes, we used the same projection (UTM, zone 34N) for all methods expect for dBBMM, as the program needed the coordinates to be in aeqd projection due to technical issues. The results from the different methods are shown in Figures 5.1 & 5.2 and are in good agreement.

According to the scientific advisor of the Society for the Protection of Prespa, George Katsadorakis, the most plausible 95% estimate of the home range of the goose appears to have been produced by the kernel density estimation with h_{ref} bandwidth (Fig. 5.1&5.2). The home range polygons generated with H_{PI} (plug-in) appeared fragmented and possibly underestimated the home range. In contrast, the 95% estimate derived from the BBMM (Fig. 5.1&5.2) appeared to overestimate the home range. Although, home ranges produced from the BBMM and the dBBMM are in different projections, the dBBMM performed slightly better than the BBMM. This result however was expected, since according to theory the dBBMM performs better than -or at least as well as- the traditional BBMM with a constant σ_m^2 (Kraustuber 2012).

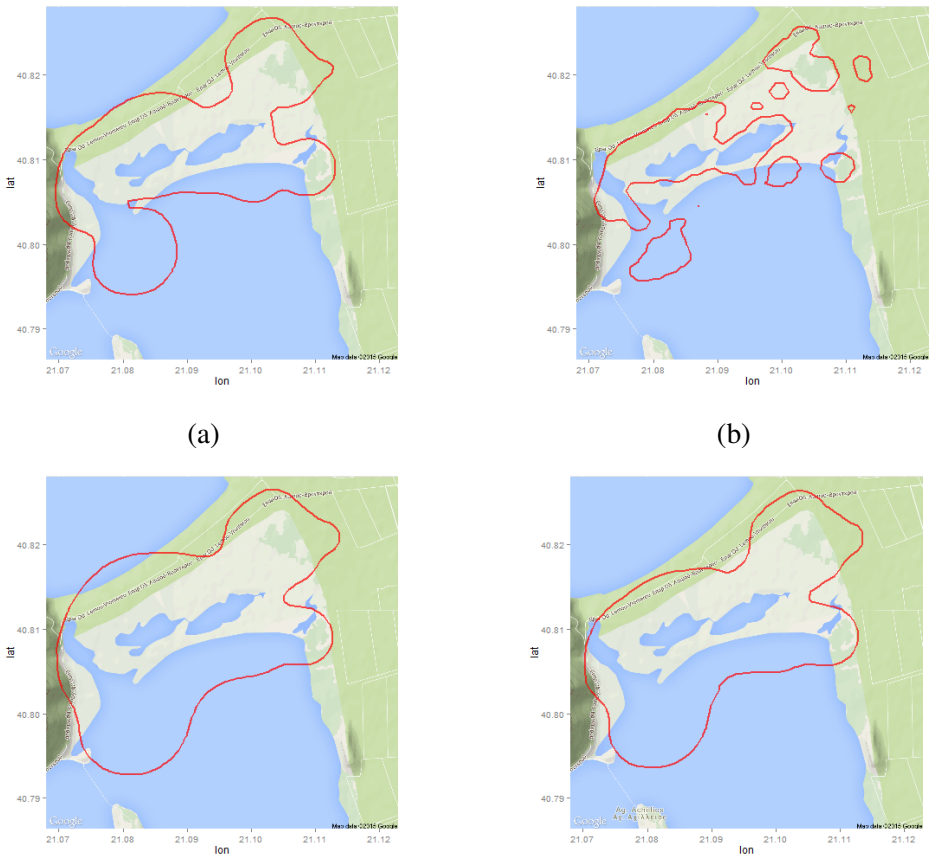


(a) (b)

(c) (d)

Figure 5.1: *Estimated home range (i.e., utilization distribution) of the greylag goose. The range in (a)&(b) is calculated using kernel density estimate with: (a) a bandwidth* $h_{ref} = 187.0615$ *and (b) a diagonal bandwidth matrix* H_{PI}*=diag(3441.018, 2502.519). The range in (c) is calculated using Brownian bridge movement method with a variance* $\sigma_m^2 = 10.1291$ *and* $\delta = 2.5$ *meters and in (d) using dynamic Brownian bridge movement method with a window size of 33 locations, margins of 13 locations and* $\delta = 2.5$ *meters. In all panels the extent is 0.2 and the grid=100.*

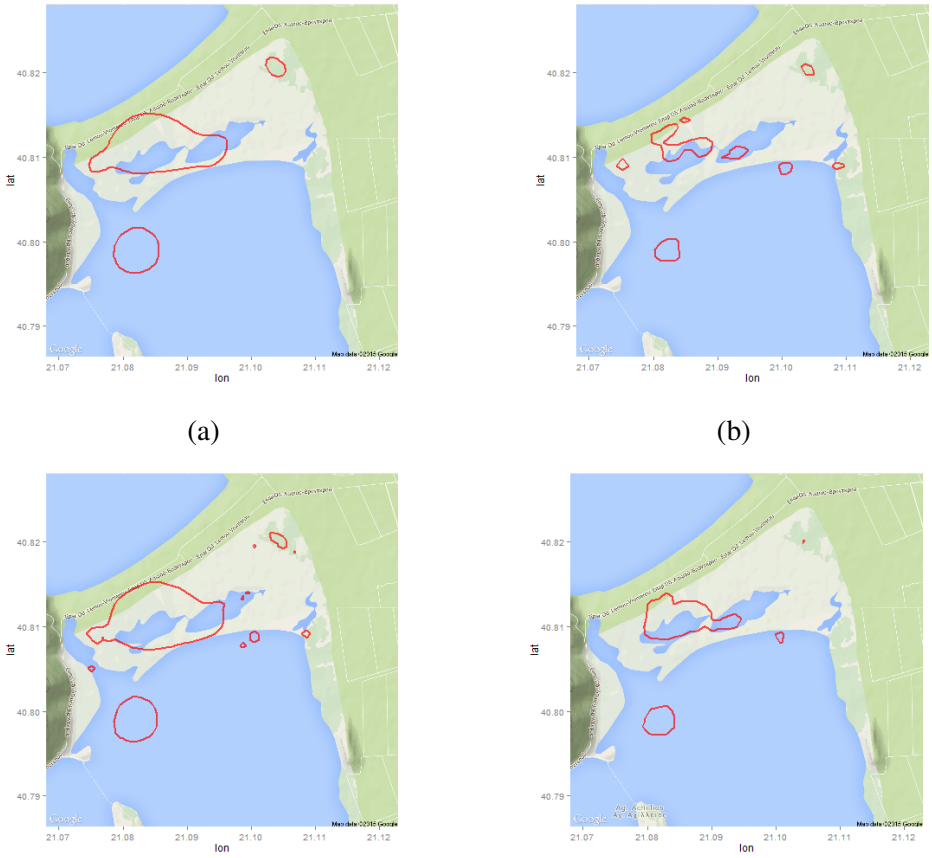
A comparison of 95% and 50% estimates of goose home range derived from the different methods are shown in Figures 5.2&5.3 respectively. In contrast to the 95% estimate derived from KDE with h_{ref} bandwidth, the 50% estimate appeared to possibly overestimate the home range. In addition, the 50% estimate produced by the dBBMM appeared to perform better than the BBMM and KDE with h_{ref} bandwidth .



(a) (b)

(c) (d)

Figure 5.2: *Comparison of 95% estimates of goose home range derived from kernel density estimation with (a) h_{ref} bandwidth selection and (b) H_{PI} bandwidth selection as well as (c) a Brownian bridge movement model and (d) a dynamic Brownian bridge movement method.*



(a) (b)

(c) (d)

Figure 5.3: *Comparison of 50% estimates of goose home range derived from kernel density estimation with (a) h_{ref} bandwidth selection and (b) H_{PI} bandwidth selection as well as (c) a Brownian bridge movement model and (d) a dynamic Brownian bridge movement method.*

5.2 Comparison by month

It is biologically valuable to compare different parts of the track with each other. For example, how do data points differ between winter, summer or between behaviors like migrating, non-migrating, resting?

We split the data into four groups based on the month in order to see how and if the home ranges differ. Another possible approach would be to split the data into two groups, one including only the months May and June, as during these months the goose appears to move less than usual since it is molting. To follow this approach it would be better if we had the data of one year or at least of six months.

In Figure 5.4 we provide some descriptive plots of the data from each month and the scatterplots of the greylag goose locations per month in google maps are provided in Figure 5.5.

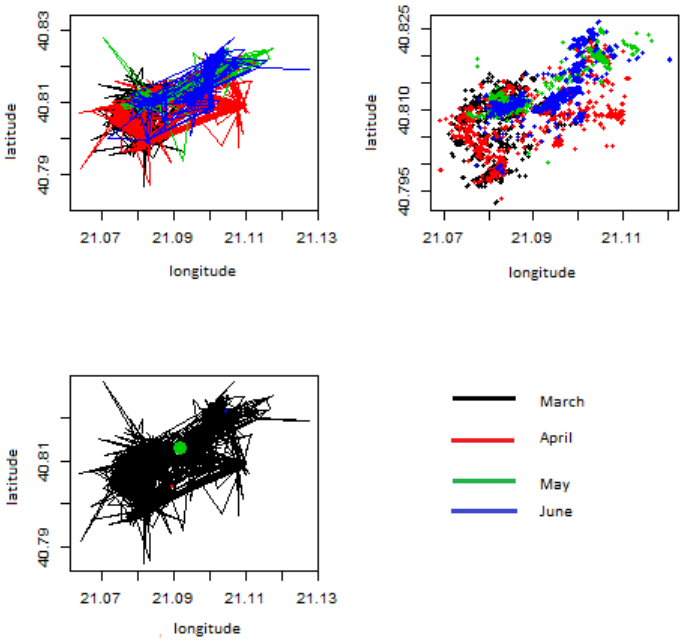
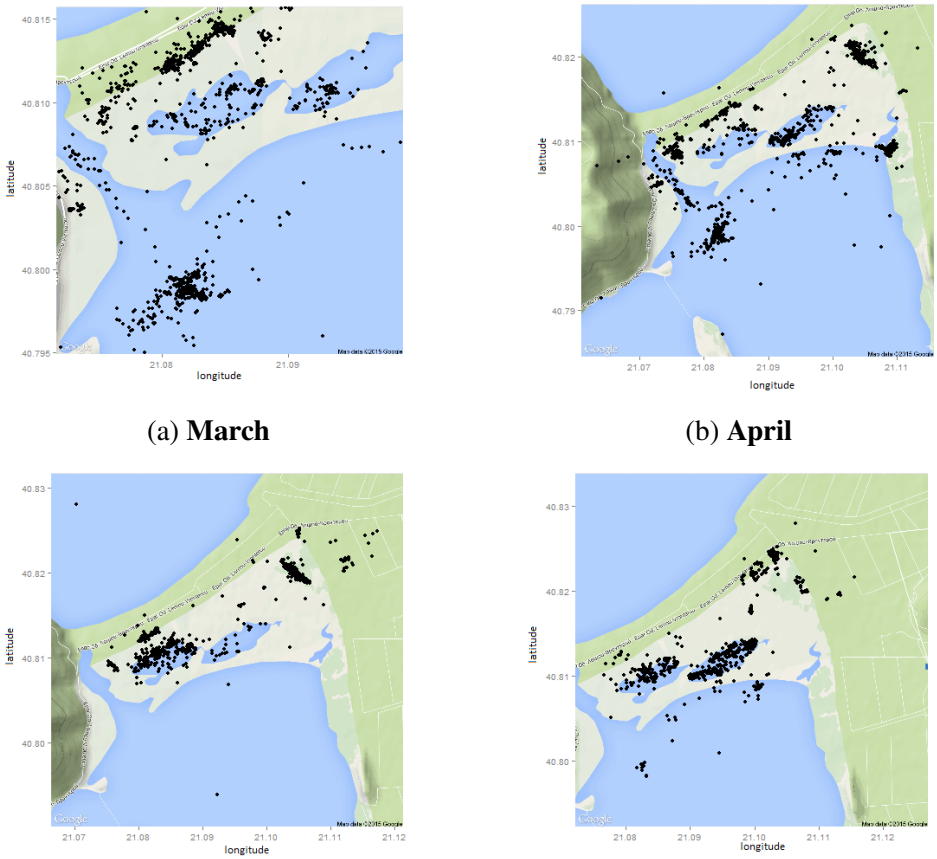


Figure 5.4: Plotting the track and adding the information for each month as coloured lines, points or circles. The colour corresponds to the month; the size of the circles represents the relative amount of time per segment.

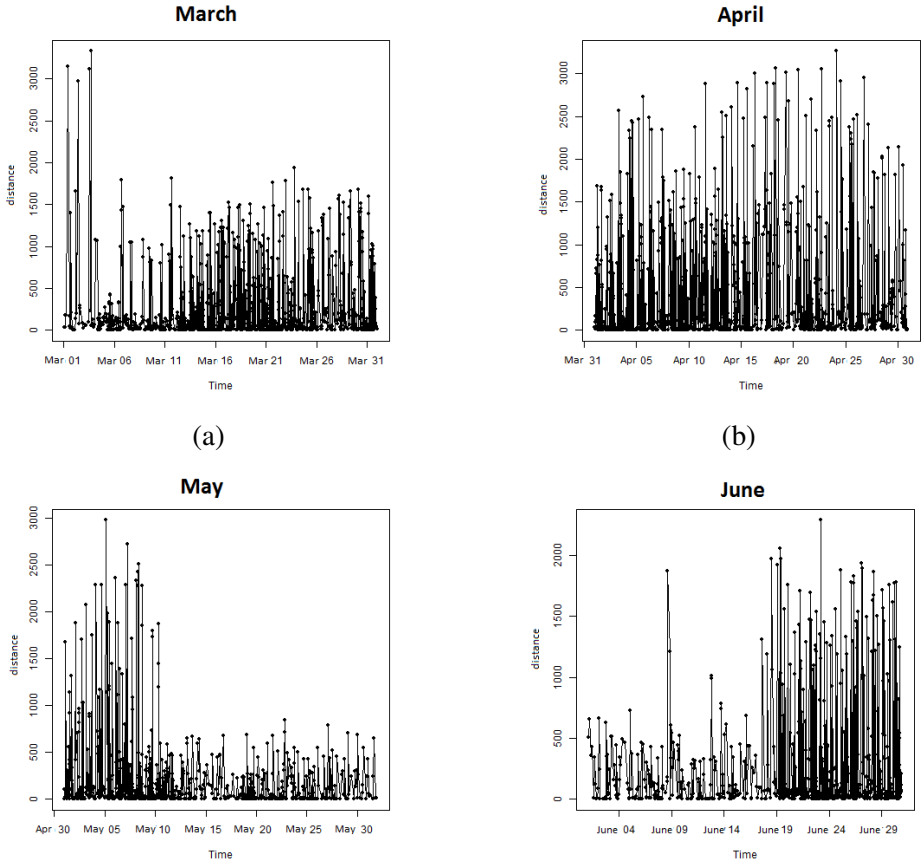


(a) **March** (b) **April**

(c) **May** (d) **June**

Figure 5.5: *Comparison of the scatterplots in google maps for (a) March (number of locations: 1440) (b) April (number of locations: 1405) (c) May (number of locations: 989) and (d) June (number of locations: 1018). Longitude/Latitude projection.*

A graphical display of the distance between two successive relocations of the goose and the time per month:



(a) (b)

(c) (d)

Figure 5.6: *Plots of the distance between two successive relocations of the greylag goose, categorised by month (a) March, (b) April, (c) May, (d) June. The distance is expressed in the units of the coordinates x,y (here in meters).*

Then we calculated the utlization distribution of the home range for each month using the KDE with bandwidths h_{ref} and H_{PI} (plug-in), the BBMM and the dBBMM. The $M(h)$ function for LSCV method again gave no minimum for none of the months.

5.2.1 March

The results from the different methods for March are shown in Figure 5.7.

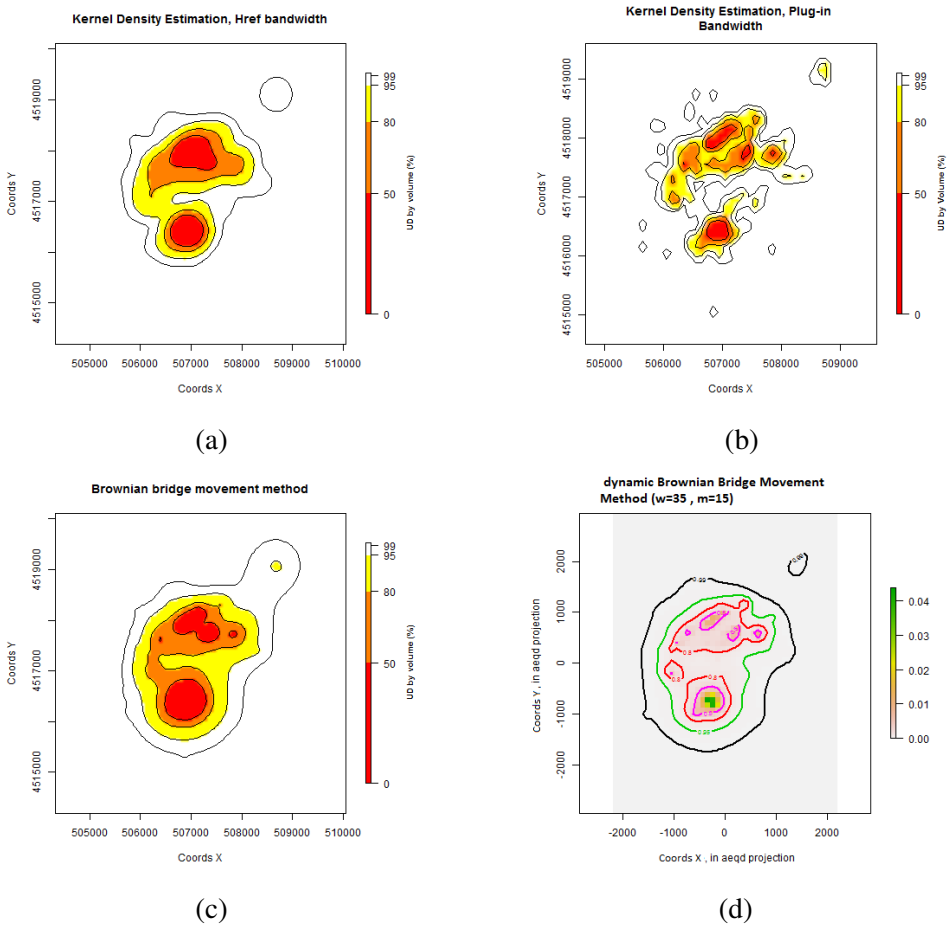


Figure 5.7: *Estimated home range (i.e., utilization distribution) of the greylag goose. The range in (a)&(b) is calculated using kernel density estimate with: (a) a bandwidth* $h_{ref} = 175.8366$ *and (b) a diagonal bandwidth matrix* H_{PI}*=diag(2734.712, 5105.481). The range in (c) is calculated using Brownian bridge movement method with a variance* $\sigma_m^2 = 8.988$ *and in (d) using dynamic Brownian bridge movement method with a window size of 35 locations and margins of 15 locations (the median of the time.lags* $= 20$ *minutes). Contours represent 50, 80, 90 and 99% of the volume of the home range estimate. In all panels the extent is 0.2 and the grid is 100.*

5.2.2 April

The results from the different methods for April are shown in Figure 5.8.

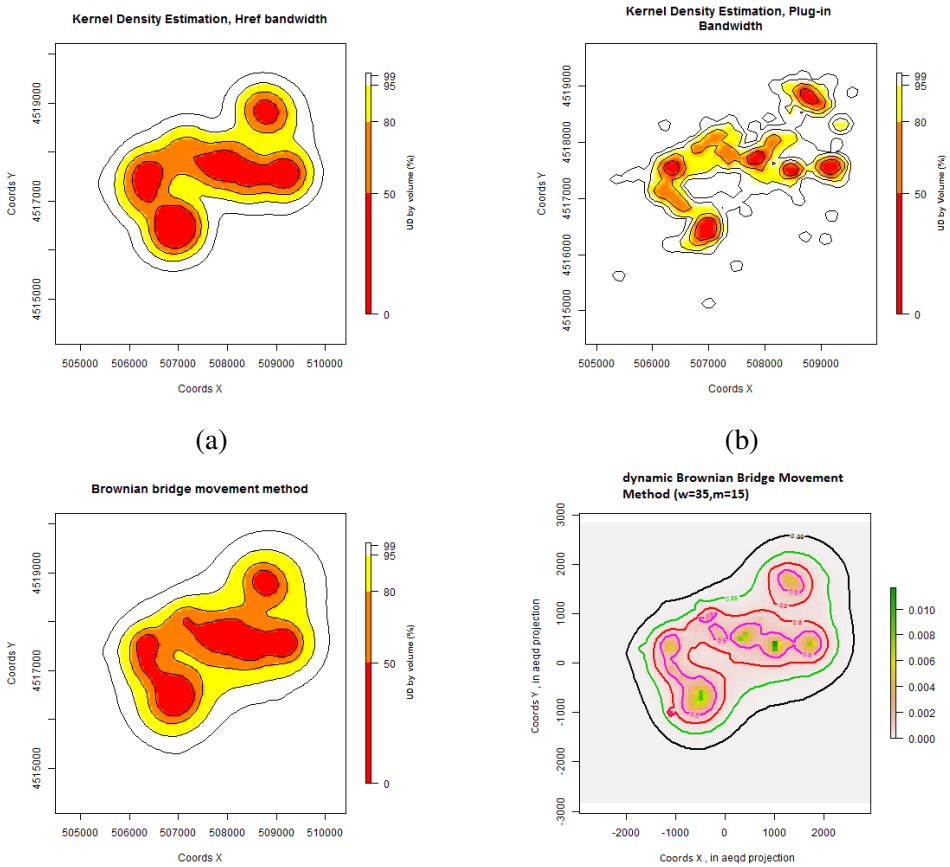


Figure 5.8: *Estimated home range (i.e., utilization distribution) of the greylag goose. The range in (a)&(b) is calculated using kernel density estimate with: (a) a bandwidth* $h_{ref} = 262.7008$ *and (b) a diagonal bandwidth matrix* H_{PI}*=diag(7792.99, 4692.919). The range in (c) is calculated using Brownian bridge movement method with a variance* $\sigma_m^2 = 13.2673$ *and in (d) using dynamic Brownian bridge movement method with a window size of 35 locations and margins of 15 locations (the median of the time.lags* ≈ 20.0333 *minutes). Contours represent 50, 80, 90 and 99% of the volume of the home range estimate. In all panels the extent is 0.2 and the grid is 100.*

5.2.3 May

The results from the different methods for April are shown in Figure 5.9.

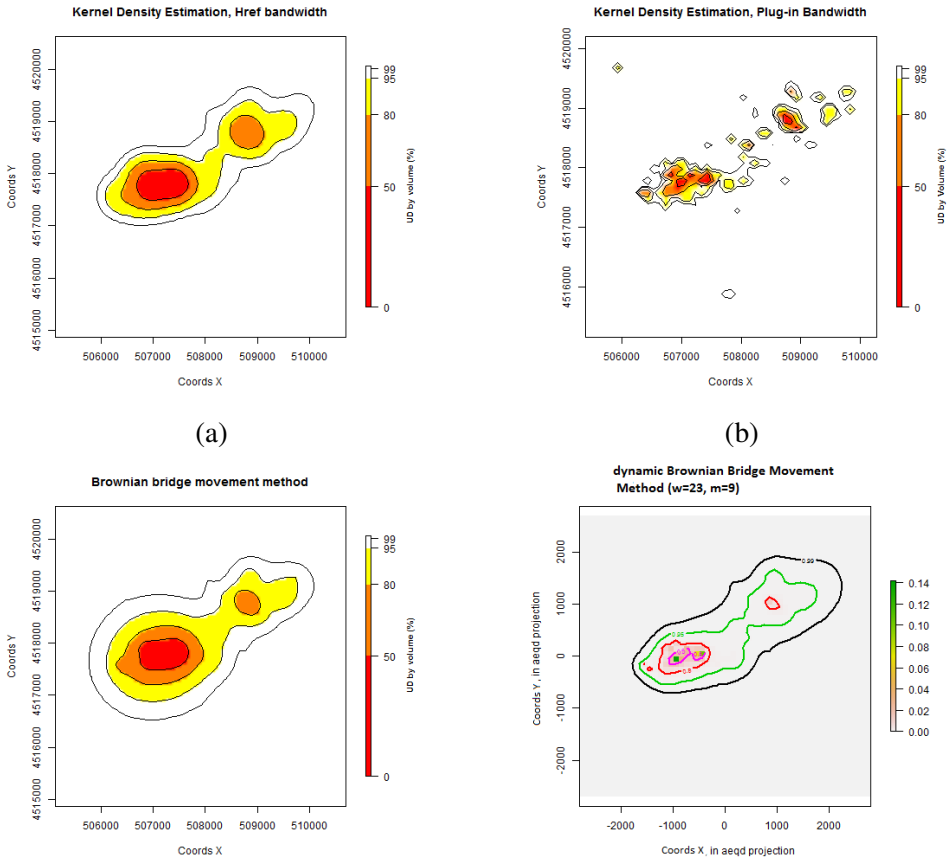


Figure 5.9: *Estimated home range (i.e., utilization distribution) of the greylag goose. The range in (a)&(b) is calculated using kernel density estimate with: (a) a bandwidth* $h_{ref} = 194.783$ *and (b) a diagonal bandwidth matrix* H_{PI}*=diag(1611.724, 560.0261). The range in (c) is calculated using Brownian bridge movement method with a variance* $\sigma_m^2 = 8.045$ *and in (d) using dynamic Brownian bridge movement method with a window size of 23 locations and margins of 9 locations (the median of the time.lags* ≈ 29.9 *minutes). Contours represent 50, 80, 90 and 99% of the volume of the home range estimate. In all panels the extent is 0.2 and the grid is 100.*

5.2.4 June

The results from the different methods for June are shown in Figure 5.10.

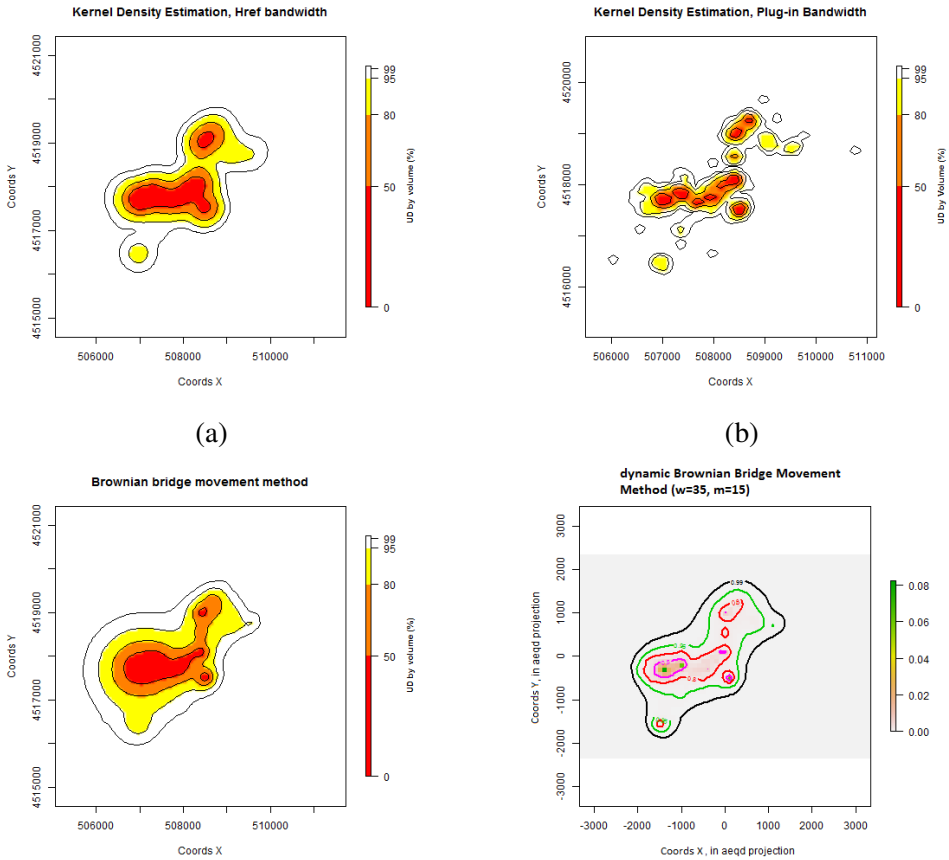


Figure 5.10: *Estimated home range (i.e., utilization distribution) of the greylag goose. The range in (a)&(b) is calculated using kernel density estimate with: (a) a bandwidth* $h_{ref} = 195.9849$ *and (b) a diagonal bandwidth matrix* H_{PI}*=diag(6139.599, 3786.221). The range in (c) is calculated using Brownian bridge movement method with a variance* $\sigma_m^2 = 8.2653$ *and in (d) using dynamic Brownian bridge movement method with a window size of 35 locations and margins of 15 locations (the median of the time.lags* ≈ 20.083 *minutes). Contours represent 50, 80, 90 and 99% of the volume of the home range estimate. In all panels the extent is 0.2 and the grid is 100.*

Chapter 6

Conclusion

Despite the fact that our data suggests that a more realistic 95% estimate of the utilization distribution may be obtained using the KDE (h_{ref}) than the BBMM, it is important to realize that there are some distinct differences in assumptions, both implicit and explicit, between the two models that can eventually lead to contradicting assessments. From our greylag goose example, it is evident that areas of frequent use were more likely to be "connected" via pathways using the BBMM as opposed to the kernel estimate. This is because the BBMM, having more of a mechanistic basis, estimates the utilization distribution by modeling the animal's expected movement path throughout an area over a period of observation. Uncertainty in the actual movement path is directly incorporated via the two ecologically based model parameters: the animal's mobility (i.e., σ_m^2) and measurable location error (Horne 2007).

In contrast, kernel smoothing techniques do not share a similar mechanistic basis. Instead, location data assumed to represent a statistical sample from some underlying probability distribution, not the animal's movement path. Location data are smoothed to an "optimal" level in order to recover, as accurately as possible, the actual underlying distribution. The value of the smoothing parameter is usually chosen based on some type of statistical procedure designed to minimize the difference between the kernel estimate and the true distribution (Horne & Garton 2006), and kernel estimates are notoriously sensitive to these values. Although the smoothing employed by kernel estimates can be viewed as an indirect method for incorporating process and measurement error into estimates of the probability of occurrence, the connection is not as direct as the BBMM and there is no connection to ecological processes (Powell 2000).

Because of the differences in fundamental assumptions, the BBMM deals with the issues of serial correlation and unequal time intervals between locations in a much more straightforward manner. Unlike other probabilistic home range models, including kernel estimates, that assume temporal independence (Worton

1987), the BBMM assumes that locations are not independent, and explicitly incorporates the time between locations into the model. The BBMM "fills in" the space between sequential locations irrespective of the density of locations, where the width of the Brownian bridge is conditioned only on the time duration between the beginning and ending locations for each pair of locations and GPS location error. As such, BBMM is able to predict movement paths that otherwise would not be observed with KDE methods (Walter 2011).

Although in theory equal time intervals between successive relocations are not a requirement of the BBMM, the method uses a Brownian bridge to estimate the probability density that the animal used any particular pixel, given its relocations. The "shape" of the Brownian bridge characterizing two successive relocations is adjusted as a function of the time lag separating these two relocations; if the time lag is short, the bridge will be narrower than if the time lag was long. Even though this approximation may be useful to account for movement constraints, e.g. an animal cannot move 20 km in two minutes, its implications may be problematic if the time lag between successive relocations is highly variable (Walter 2011).

The variability in time lag between successive locations was important in our data-set and the BBMM method resulted in producing larger home range size than the KED with h_{ref}, which fails to take into consideration the time lag (Fig. 5.2). Time lag was important for greylag goose (Fig. 4.2) as the solar-powered GPS collar or occupied habitats prevented the battery from maintaining a full charge, thereby not allowing GPS data logging upon fix attempt (Cappelle 2011). According to Walter (2011) a possible way to improve our results, would be a "crude" approach to eliminate the top 1% of outliers of time difference such that 99% of the original data would be included (O_{REM}). While use of O_{REM} may seem reasonable for some species and studies in order to eliminate large time differences and thus, resulting in tighter home ranges, Walter (2011) cautions researchers from using such an approach without considering its implications to the ecological questions at hand and prior to determining distribution of time differences with locations from each study animal.

Appendices

Appendix A

Code in R

A.1 Kernel estimation

```
# Read the data
 data_geese<-read.csv2("C:Users/xristina/Desktop/H01 logger1(3-6).csv",header =
     TRUE, sep =";", quote = "\"",dec=",")
 head(data\_geese)

# Get only the coordinates in degrees
 library(rgdal)
 loc<-cbind("x"=data_geese$LONDEG,"y"=data_geese$LATDEG)

#Define the projection of the coordinates (re-express the coordinates in meters)
#Here in UTM, zone 34N
 loc_meters<-project(loc, "+proj=utm +zone=34N ellps=WGS84")

#Make spatial points data frame using xy,attributes and projection
 proj4string<-CRS("+proj=utm +zone=34N ellps=WGS84")
 spdf<-SpatialPointsDataFrame(loc_meters,data_geese,proj4string=proj4string)

#Create spatial Points from just the xy's
 loc.pts<-SpatialPoints(loc_meters,proj4string=proj4string)
```

A.1.1 H_{ref}

```
library(adehabitatHR)

# Calculate the UD using KED, href
##kernel=bivnorm ,by default extent=1 ,grid=60
 kud1<-kernelUD(loc.pts , h="href")
 kud1
 kud1@h
 image(kud1)

##Controlling the grid
#The UD is estimated at the center of each pixel of a grid. Although the size and
     resolution of the grid doesn't have a large effect on the estimates (
    Silverman 1986), it is sometimes useful to be able
```

```
to control the parameters defining this grid.
## The parameter grid controls the resolution of the grid and the parameter
     extent controls its extent.

### Graphical parameters
 par(mar=c(0,0,2,0))
 par(mfrow=c(2,2))
#Estimation of the UD with grid=80 and extent=0.1
 image(kernelUD(loc.pts, grid=80, extent=0.1))
 title(main="grid=80, extent=0.1")
#Estimation of the UD with grid=100 and extent=0.1
 image(kernelUD(loc.pts, grid=100, extent=0.1))
 title(main="grid=100, extent=0.1")
#Estimation of the UD with grid=80 and extent=0.2
 image(kernelUD(loc.pts, grid=80, extent=0.2))
 title(main="grid=80, extent=0.2")
#Estimation of the UD with grid=100 and extent=0.2
 image(kernelUD(loc.pts, grid=100, extent=0.2))
 title(main="grid=100, extent=0.2")

#By eye I choose that  grid=100 & extent=0.2 fit better the data.
#Calculate again the kudl
 kud1←kernelUD(loc.pts , h="href", grid=100, extent=0.2)
 kud1
 kud1@h
 image(kud1)

##Try a different kernel: "Epanechnikov"
 kud2←kernelUD(loc.pts, h="href",kern=c("epa"), grid=100,extent=0.2)
 kud2
 image(kud2)
 kud2@h
#Compare the 2 kernels
  par(mar=c(0,0,2,0))
  par(mfrow=c(2,1))
  image(kud1)
   title(main="kernel=bivnorm, grid=100, extent=0.2")
  image(kud2)
   title(main="kernel=epa ,grid=100, extent=0.2")
#They are almost the same, so for simplicity use the bivariate normal kernel.

#Estimating the home range from the UD
#The home range deduced from the UD as the minimum area on which the probability
     to relocate the animal is equal to a specified value  e.x. 0.95 (95%).

#The home range in vector mode
 homerange1←getverticeshr(kud1,percent=95,unin=c("m"),unout=c("km2"),standardize
     =T)
 class(homerange1)
 plot(homerange1, col=1:4)

#The homerange in raster mode
#This function (getvolumeUD) modifies the UD component of the object passed as
     argument, so that the value of a pixel is equal to the percentage of the
     smallest home range containing this pixel.
 vud←getvolumeUD(kud1,standardize=T)
 vud

#To see the difference between the output kernelUD and getvolumeUD
 par(mfrow=c(2,1))
 par(mar=c(0,0,2,0))
#The output of the kernel UD
```

```
 image(kud1)
 title("output of kernelUD")
#Convert into a suitable data structure for the use of the contour
 xyz<-as.image.SpatialGridDataFrame(kud1)
 contour(xyz,add=TRUE)
#and similarly for the output of the getvolumeUD
 par(mar=c(0,0,2,0))
 image(vud)
 title("output of getvolumeUD")
 xyzv<-as.image.SpatialGridDataFrame(vud)
 contour(xyzv,add=TRUE)

#The output of getvolumeUD can be used to compute the home range (the labels of
    the contour lines correspond to the home ranges computed with various
    probability levels).

#For example, to get the rasterized 95% home range of the goose.
#Store the volume uder the UD of the fisrt animal in fud.
 fud<-vud
#Store the value of the volume under UD in a vector.
 hr95<-as.data.frame(fud)[,1]
#if the hr95<=95 then the pixel belongs to the home range and takes 1, 0
    otherwise
 hr95<-as.numeric(hr95 <= 95)
 hr95<-data.frame(hr95)
#Convert into SPixelsDF
 coordinates(hr95)<-coordinates(vud)
 gridded(hr95)<-TRUE
#Dispaly the results
 image(hr95)

#The home range size
 as.data.frame(homerange1)

#The home range size  for several probability levels.
 ii<-kernel.area(kud1, percent=seq(50,95,by=5),unin=c("m"),unout=c("km2"),
     standardize=T)
 ii
 plot(ii)

#Create a plot in raster mode for several probability levels
 library(raster)
 plot.new()
 breaks<-c(0,50,80,95,99)
 hm99<-getverticeshr(kud1,percent=99,ida=NULL,unin="m",unout="km2",stadardize=T)
 hm80<-getverticeshr(kud1,percent=80,ida=NULL,unin="m",unout="km2",stadardize=T)
 hm95<-getverticeshr(kud1,percent=95,ida=NULL,unin="m",unout="km2",stadardize=T)
 hm50<-getverticeshr(kud1,percent=50,ida=NULL,unin="m",unout="km2",stadardize=T)
 hvol<-getvolumeUD(kud1,standardize=T)
 hvol.raster<-raster(hvol)
 plot(hvol.raster,col=heat.colors(3),breaks=breaks,interpolate=T,main="Kernel
     Density Estimation, Href bandwidth",xlab="Coords X",ylab="Coords Y",legend.
     shrink=0.8,legend.args=list(text="UD by volume(%)",side=4,font=2,line=2.5,
     cex=0.8))
 plot(hm50,add=T)
 plot(hm80,add=T)
 plot(hm95,add=T)
 plot(hm99,add=T)
 points(loc_meters,pch=1,cex=0.5)

 #Contours in google maps
 library(rgdal)
```

```
 library(ggplot2)
 library(maptools)
 hm95<-getverticeshr(kud1,percent=95,ida=NULL,unin="m",unout="km2",stadardize=T)
#Save a shapefile
 writeOGR(hm95, dsn="C:/Users/xristina/Desktop",layer="hm95", driver="ESRI
     Shapefile")
#Read shapefile
 shapefile<-readShapeSpatial('C:/Users/xristina/Desktop/hm95', proj4string = CRS(
     "+proj=utm +zone=34N ellps=WGS84"))
#Transform the projection
 shp<- spTransform(shapefile, CRS("+proj=longlat +datum=WGS84"))
 data<- fortify(shp)
 library(ggmap)
 library(mapproj)
 library(raster)
 locs_df<-as(loc.pts,"data.frame")
 m<-get_map(location=bbox(extent(loc)),source="google")
 ggmap(m)+ geom_polygon(aes(x=long,  y=lat), data = data,colour = 'red', fill = '
     black', alpha = .4, size = .3)
```

A.1.2 H_{LSCV}

```
#Read the data
 data_geese<-read.csv2("C:/Users/xristina/Desktop/H01 logger1(3-6).csv",header =
     TRUE, sep = ";", quote = "\"",dec = ",")
 head(data_geese)
#Get only the coordinates in degrees
 library(rgdal)
 loc<-cbind("x"=data_geese$LONDEG,"y"=data_geese$LATDEG)
#Define the projection of the coordinates (re-express the coordinates in meters)
 loc_meters<-project(loc, "+proj=utm +zone=34N ellps=WGS84")
#Make spatial points data frame using xy,attributes and projection
 proj4string<-CRS("+proj=utm +zone=34N ellps=WGS84")
 spdf<-SpatialPointsDataFrame(loc_meters,data_geese,proj4string=proj4string)
#Create spatial Points from just the xy's
 loc.pts<-SpatialPoints(loc_meters,proj4string=proj4string)

#Calculate the UD using Hlscv
 library(adehabitatHR)
 kud3<-kernelUD(loc.pts, h="LSCV",grid=100, extent=0.2)
 plotLSCV(kud3)
```

A.1.3 $H_{plug-in}$

```
 library(ks)
 library(rgdal)
 library(maptools)
 library(gpclib)
 library(PBSmapping)
 library(adehabitat)
 library(adehabitatHR)
 library(raster)

#Need to calculate the bandwidth matrix to use later in creating the KDE.
 Hpi1<- Hpi.diag(x = loc_meters)
```

```
 Hpi1

##Create spatial points from just the xy's
 loc.pts<- SpatialPoints(loc, proj4string=proj4string)

#For home range calculations, some packages require evaluation points (ks) while
    others require grid as spatial pixels (adehabitatHR).

##Set the expansion value for the grid and get the bbox from the
    SpatialPointsDataFrame
 expandValue<-500 #This value is the amount to add on each side of the bbox
 boundingVals<- spdf@bbox
##Get the change in x and y and adjust using expansion value.
 deltaLong<- as.integer(((boundingVals[1,2]) - (boundingVals[1,1])) + (2*
     expandValue))
 deltaLat<- as.integer(((boundingVals[2,2]) - (boundingVals[2,1])) + (2*
     expandValue))
##100 meters grid for data
 gridRes<- 100
 gridSizeX<- deltaLong / gridRes
 gridSizeY<- deltaLat / gridRes
##Offset the bounding coordinates to account for the additional area
 boundingVals[2,1]<- boundingVals[2,1] - expandValue
 boundingVals[2,2]<- boundingVals[2,2] + expandValue
 boundingVals[1,1]<- boundingVals[1,1] - expandValue
 boundingVals[1,2]<- boundingVals[1,2] + expandValue
##Grid Topology object is basis for sampling grid (offset, cellsize, dim).
 gridTopo<-GridTopology((boundingVals[,1]),c(gridRes,gridRes),c(gridSizeX,
     gridSizeY))
##Using the Grid Topology and projection create a SpatialGrid class.
 sampGrid<- SpatialGrid(gridTopo, proj4string = proj4string)
##Cast over to Spatial Pixels
 sampSP <- as(sampGrid, "SpatialPixels")
##Convert the SpatialGrid class to a raster
 sampRaster<- raster(sampGrid)
##Set all the raster values to 1 such as to make a data mask.
 sampRaster[] <- 1
##Get the center points of the mask raster with values set to 1.
 evalPoints <- xyFromCell(sampRaster, 1:ncell(sampRaster))

#Here we can see how grid has a buffer around the locations and trajectory. This
    will ensure that we project our home range estimates into a slightly larger
    extent than the original points extent (bbox) alone.

 plot(sampRaster)
 points(loc_meters, pch=1, cex=0.5)

##Create the KDE using the evaluation points.
 hpikde <- kde(x=loc_meters, H=Hpi1, eval.points=evalPoints)

#Create a template raster based upon the mask and then assign the values from the
    kde to the template
 hpikde.raster <- raster(sampRaster)
 hpikde.raster <- setValues(hpikde.raster, hpikde$estimate)

##Lets take this raster and put it back into an adehabitat object. This is
    convenient to use other adehabitat capabilities such as overlap indices or
    percent volume contours.
##Cast over to SPxDF
 hpikde.px <- as(hpikde.raster, "SpatialPixelsDataFrame")
##Create new estUD using the SPxDF
 hpikde.ud <- new("estUD", hpikde.px)
```

```
##Assign values to a couple slots of the estUD
 hpikde.ud@vol = FALSE
 hpikde.ud@h$meth = "Plug-in Bandwidth"
##Convert the UD values to volume using getvolumeUD from adehabitatHR and cast
    over to a raster.
 hpikde.ud.vol <- getvolumeUD(hpikde.ud, standardize=TRUE)
 hpikde.ud.vol.raster <- raster(hpikde.ud.vol)

##Here we generate volume contours using the UD.
 hpikde.99vol <- getverticeshr(hpikde.ud, percent = 99,ida = NULL, unin = "m",
     unout = "km2", standardize=TRUE)
 hpikde.80vol <- getverticeshr(hpikde.ud, percent = 80, ida = NULL, unin = "m",
     unout = "km2", standardize=TRUE)
 hpikde.95vol <- getverticeshr(hpikde.ud, percent = 95,ida = NULL, unin = "m",
     unout = "km2", standardize=TRUE)
 hpikde.50vol <- getverticeshr(hpikde.ud, percent = 50,ida = NULL, unin = "m",
     unout = "km2", standardize=TRUE)

#Let's put the HR volume contours
 plot.new()
 breaks <- c(0, 50, 80, 95, 99)
 plot(hpikde.ud.vol.raster, col=heat.colors(3), breaks=breaks,interpolate=TRUE,
     main="Kernel Density Estimation, Plug-in Bandwidth",xlab="Coords X", ylab="
     Coords Y", legend.shrink=0.80, legend.args=list(text="UD by Volume (%)",
 side=4, font=2, line=2.5, cex=0.8))
 plot(hpikde.50vol, add=TRUE)
 plot(hpikde.80vol, add=TRUE)
 plot(hpikde.95vol, add=TRUE)
 plot(hpikde.99vol, add=TRUE)
 points(loc_meters, pch=1, cex=0.5)

# Contours in google maps
 library(rgdal)
 library(ggplot2)
 library(maptools)
 hp95 <- getverticeshr(hpikde.ud, percent =95,ida = NULL, unin = "m", unout = "
     km2", standardize=TRUE)
#Save a shapefile
 writeOGR(hp95, dsn="C:/Users/xristina/Desktop", layer="hp95", driver="ESRI
     Shapefile")
#Read shapefile
 shapefile <- readShapeSpatial('C:/Users/xristina/Desktop/hp50',   proj4string =
     CRS("+proj=utm +zone=34N ellps=WGS84"))
#Transform the projection
 shp <- spTransform(shapefile, CRS("+proj=longlat +datum=WGS84"))
 data <- fortify(shp)

 library(ggmap)
 library(mapproj)
 library(raster)
m<-get_map(location=bbox(extent(loc)),source="google")
 ggmap(m) + geom_polygon(aes(x=long,  y=lat), data = data,colour = 'red', fill =
     'black', alpha = .4, size = .3)
```

A.2 Brownian bridge movement method

```
#Read the data
 data_g←read.csv2("C:/Users/xristina/Desktop/H01 logger2.csv", header = TRUE,
     sep = ";", quote = "\"", dec = ",")
 head(data_g)
 library(rgdal)
#Get only the coordinates in degrees
 loc←cbind("x"=data_g$LONDEG, "y"=data_g$LATDEG)
#Define the projection of the coordinates (re-express the coordinates in meters)
 loc_meters←project(loc, "+proj=utm +zone=34N ellps=WGS84")
 locs←as.data.frame(loc_meters)
 names(locs)←c("x","y")

#Transform into an object of the class POISIXct (store date and time in R).
 da←as.character(data_g$DATE.TIME)
 head(da)
 da←as.POSIXct(strptime(as.character(data_g$DATE.TIME),format="\%y\%m\%d \%H\%M
     \%S"),tz="UTC")
#Needs id, since we have only one animal id=1.
 idd←rep(1,length(locs$x))
#We can create an object of ltraj to store the goose's movements
 libary(adehabitatHR)
 data_ge←as.ltraj(xy=locs[,c("x","y")],date=da, id=idd , typeII=TRUE)
 data_ge
 plot(data_ge,xlab="coordinates x in metres,in UTM zone",ylab="coordinates y in
     metres,in UTM zone")
 plotltr(data_ge)

#Brownian Bridge kernel method
#Suppose sig2=2.5 meters (sd=2.5)
 lik1←liker(data_ge, sig2=2.5, rangesig1=c(10,100))
#Try a different range for sig1
 lik←liker(data_ge, sig2=2.5, rangesig1=c(1,20))
#Calculate the UD using BBMM
 tata←kernelbb(data_ge,sig1=10.1291,sig2=2.5, grid=100, extent=0.2)
 tata
 image(tata)
 plot(getverticeshr(tata,95),add=TRUE, lwd=2)

#Create a plot in raster mode for several probability levels.
 library(raster)
 plot.new()
 breaks←c(0,50,80,95,99)
 hm99←getverticeshr(tata,percent=99,ida=NULL,unin="m",unout="km2",stadardize=T)
 hm80←getverticeshr(tata,percent=80,ida=NULL,unin="m",unout="km2",stadardize=T)
 hm95←getverticeshr(tata,percent=95,ida=NULL,unin="m",unout="km2",stadardize=T)
 hm50←getverticeshr(tata,percent=50,ida=NULL,unin="m",unout="km2",stadardize=T)
 hvol←getvolumeUD(tata,standardize=T)
 hvol.raster←raster(hvol)
 plot(hvol.raster,col=heat.colors(3),breaks=breaks,interpolate=T,main="Brownian
     bridge movement method", xlab="Coords X",ylab="Coords Y", legend.shrink=0.8,
      legend.args=list(text="UD by volume (%)", side=4, font=2,line=2.5,cex=0.8))
 plot(hm50,add=T)
 plot(hm80,add=T)
 plot(hm95,add=T)
 plot(hm99,add=T)
 points(loc_meters,pch=1,cex=0.5)

#Contours in google maps (95%)
 library(rgdal)
```

```
 library(ggplot2)
 library(maptools)
 hm95<-getverticeshr(tata,percent=95,ida=NULL,unin="m",unout="km2",stadardize=T)
#Save a shapefile
  writeOGR(hm95, dsn="C:/Users/xristina/Desktop", layer="hm95",driver="ESRI
      Shapefile")
#Read the shapefile
 shapefile<-readShapeSpatial('C:/Users/xristina/Desktop/hm95',proj4string = CRS("
     +proj=utm +zone=34N ellps=WGS84"))
#Transform the projection
 shp<- spTransform(shapefile, CRS("+proj=longlat +datum=WGS84"))
 data<- fortify(shp)

 library(ggmap)
 library(mapproj)
 library(raster)
 m<-get_map(location=bbox(extent(loc)),source="google")
 ggmap(m) + geom_polygon(aes(x=long,  y=lat), data = data,colour = 'red', fill =
     'black', alpha = .4, size = .3)
```

A.3 Dynamic Brownian bridge movement method

```
# Read the data
 data_g<-read.csv2("C:/Users/xristina/Desktop/H01 logger2.csv", header = TRUE,
     sep = ";", quote = "\"",dec = ",")
 head(data\_g)
 library(move)
 goose<-move(x=data_g$LONDEG, y=data_g$LATDEG, data=data_g,time=as.POSIXct(
     strptime(as.character(data_g$DATE.TIME), format="\%y\%m\%d \%H\%M\%S"),tz="
     UTC"),proj=CRS("+proj=longlat +ellps=WGS84+datum=WGS84"),animal=rep(1,length
     (locs$x)),sensor=rep("gps",length(locs$x)))

#Show and summarize
 show(goose)
 summary(goose)
 plot(goose)
 head(as.data.frame(goose))
 head(timestamps(goose))
 n.locs(goose)
 head(time.lag(goose, units="mins"))
 hist(time.lag(goose ,units='mins'))
 idData(goose)
 as(goose,'data.frame')
 speed(goose)
 hist(speed(goose))
 distance(goose) # measured in meters
 hist(distance(goose))
 plot(goose,type="o",col=4,lwd=2,pch=20,xlab="location_long",ylab="location_lat")
 plot(goose)
 lines(goose)

#Scatterplot in google maps
 coords<-cbind(data_g$LONDEG,data_g$LATDEG)
 head(coords)
 require(ggmap)
 require(mapproj)
 goose_df<-as(goose,"data.frame")
```

```
 m ← get_map(location=bbox(extent(coords)), source="google")
 ggmap(m)+geom_path(data=goose_df,aes(x=data_g$LONDEG,y=data_g$LATDEG))

#Dynamic Brownian Bridge Movement Model
 proj4string(goose)
 g_aeqd←spTransform(x=goose, CRSobg="+proj=aeqd", center=T)
 proj4string(g_aeqd)
#Window size=23, margin = 9, location error=2.5 meters
 dBMvar ← brownian.motion.variance.dyn(object=g_aeqd, location.error=rep(2.5,n.
     locs(g_aeqd)),margin=9, window.size=23)
 getMotionVariance(dBMvar)
 plot(timestamps(g_aeqd), getMotionVariance(dBMvar),type='s')
 g_dbbmm←brownian.bridge.dyn(g_aeqd, location.error=rep(2.5,n.locs(g_aeqd)),
     margin=9, window.size=23,raster=100,ext=0.2)
 plot(g_dbbmm, xlab="Longtitude, in aeqd projection", ylab="Latitude, in aeqd
     projection")
 contour(g_dbbmm,levels=c(.5,.8,.95,.99),col=c(6,2,3,9), add=T, lwd=2)

#Calculating UD area size
#for example 95%
 gb←getVolumeUD(g_dbbmm)
#Cells that belong to the contour will get the value 1 while others get 0
 gb←gb<=0.95
 area←sum(values(gb))
#The area is the number of the cell multiplied by the actual size of the  of the
    raster cells(100 metres here).
 area
 image(gb)

#Contours in google maps
 library(rgdal)
 library(ggplot2)
 library(maptools)
 x←raster2contour(g_dbbmm,level=c(.95))
#Save a shapefile
  writeOGR(x, dsn="C:/Users/xristina/Desktop", layer="hm95", driver="ESRI
      Shapefile")
#Read shapefile
 shapefile ← readShapeSpatial('C:/Users/xristina/Desktop/hm95', proj4string =
     CRS("+proj=aeqd +ellps=WGS84 +lon_0=21.09547 +lat_0=40.80724 "))
#Transform the projection
 shp ← spTransform(shapefile, CRS("+proj=longlat +datum=WGS84"))
 data ← fortify(shp)

 library(ggmap)
 library(mapproj)
 library(raster)
 locs_df←as(loc.pts,"data.frame")
 m←get_map(location=bbox(extent(loc)),source="google")
 ggmap(m) + geom_polygon(aes(x=long,  y=lat), data = data,colour = 'red', fill =
     'black', alpha = .4, size = .3)
```

Appendix B

Universal Transverse Mercator (UTM)

The UTM system applies the Transverse Mercator projection to mapping the world, using 60 pre-defined standard zones to supply parameters. UTM zones are six degrees wide. Each zone exists in a North and South variant.

Universal Transverse Mercator	The Northern Hemisphere projections for the infamous UTM system consisting of 120 zones (60 different zones with North and South variants of each).Originally developed for military use and now widely misused in civil mapping.
Universal Transverse Mercator (South)	The Southern Hemisphere projections for UTM. These are mainly distinguished by each having a Northing parameter of 10 million so that no coordinates need involve negative numbers.

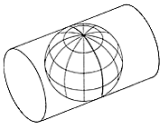
Limitations

The accuracy of any Transverse Mercator projection quickly decreases from the central meridian. Therefore, it is strongly recommended to restrict the longitudinal extent of the projected region when using Universal Transverse Mercator projections to +/- 6 degrees from the central meridian.

The **Mercator** projection maps the world onto a cylinder where the central ring of tangency is the Earth's Equator.

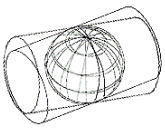
Near the Equator, the Mercator projection provides low distortion. Away from the Equator distortion becomes very high. This limits the utility of the Mercator

projection to regions near the Equator. That is a big limitation because most places that people live (and thus, most of the regions that people most frequently map) are located not along the Equator but along North-South directions, such as from North America to South America.



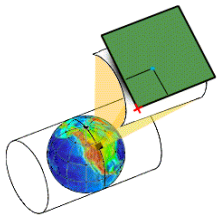
Turning the Mercator projection's cylinder so that it is tangent to the Earth along a meridian (longitude line) instead of the Equator results in what is called a **Transverse Mercator** projection. If we created a Transverse Mercator projection that had a meridian as the central ring of the cylinder we could make local maps anywhere along the North-South line of tangency. If the maps are limited to the thin, vertical region near the meridian of tangency they will be relatively free of distortion.

The problem is that any Transverse Mercator projection created by choosing any one meridian as a line of tangency is useful only near that meridian. If we pick a North-South line running through Athens we can make maps all the way from Scandinavia down the length of Africa, but any maps using this projection in North and South America would be hopelessly distorted.



The **Universal Transverse Mercator** system of projections deals with this by defining 60 different standard projections, each one of which is a different Transverse Mercator projection that is slightly rotated to use a different meridian as the central line of tangency. Each different center-line defines a **UTM Zone**. The "UTM Zone" is a shorthand way of naming a specific, different projection that consists of a Transverse Mercator projection using a different meridian as the center-line. By rotating the cylinder in 60 steps (six degrees per step) UTM

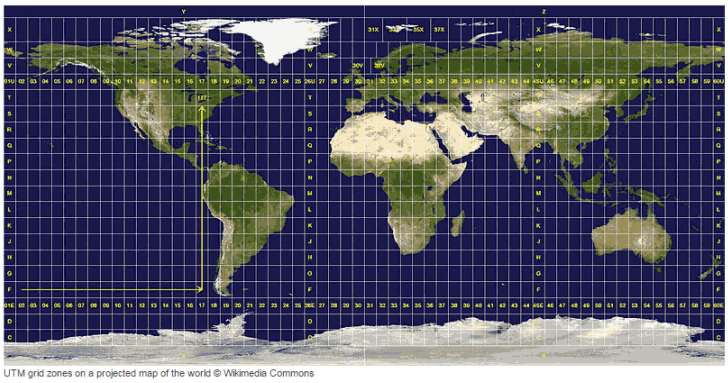
assures that all spots on the Earth will be within 3 degrees of the center-line of one of the 60 cylindrical projections.



Each UTM Zone is a Different Projection using a different system of coordinates.Therefore, if someone attempt to "combine" different UTM zones into a map that is projected using only one of those UTM zones, will result in distortion in the locations and shapes of the objects that originated in a different zone map. Geographic shapes that look good in a transverse Mercator projection centered upon a given UTM zone line will be very distorted when illustrated in a UTM projection centered upon a different zone line.

If we need to combine objects from several different UTM zones, the correct solution is to choose a different projection (such as a conic or azimuthal projection) for the combined map that provides low distortion over the entire region of interest.

Remember, although no projection is perfect for all uses some projections are better than others in the uses for which they were designed. UTM was designed to map objects within one zone at a time. It is a very bad choice if objects from several zones must be shown together on the same map.



UTM grid zones on a projected map of the world © Wikimedia Commons

Within each longitudinl zone the transverse mercator projection is used to give co-ordinates (eastings and northings) in meters. Âă For the eastings, the origin is defined as a point 500,000 meters west of the central meridian of each longitudinal zone, giving an easting of 500,000 meters at the central meridian.

For the northings in the northern hemisphere, the origin is defined as the equator.

For the northings in the southern hemisphere, the origin is defined as a point 10,000,000 meters south of the equator.Âă

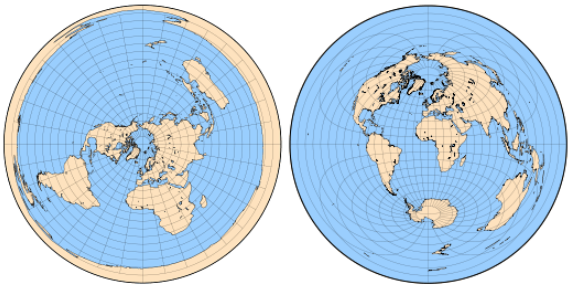
The co-ordinates thus derived define a location within a UTM projection zone either north or south of the equator, but because the same co-ordinate system is repeated for each zone and hemisphere, it is necessary to additionally state the UTM longitudinal zone and either the hemisphere or latitudinal zone to define the location uniquely world-wide.

`http://www.georeference.org/`
`http://www.dmap.co.uk/`

Appendix C

Azimuthal equidistant projection

TheÂă*azimuthal equidistant projection* is an azimuthal map projection. It has the useful properties that all points on the map are at proportionately correct distances from the center point, and that all points on the map are at the correct azimuth (direction) from the center point. A useful application for this type of projection is a polar projection which shows all meridians (lines of longitude) as straight, with distances from the north pole represented correctly.



North polar and equatorial aspects of azimuthal equidistant projection

Name	Azimuthal Equidistant
EPSG Code	
GeoTIFF Code	CT_AzimuthalEquidistant (12)
OGC WKT Name	Azimuthal_Equidistant
Supported By	GeoTIFF, PROJ.4, OGC WKT

Projection Parameters

Name	EPSG #	GeoTIFF ID	OGC WKT	Units	Notes
Latitude of projection center	1	ProjCenterLat	latitude_of_center	Angular	
Longitude of projection center	2	ProjCenterLong	longitude_of_center	Angular	
False Easting	6	FalseEasting	false_easting	Linear	
False Northing	7	FalseNorthing	false_northing	Linear	

+proj=aeqd +lat_0=*Latitude at projection center*
+lon_0=*Longitude at projection center*
+x_0=*False Easting*
+y_0=*False Northing*

http://www.remotesensing.org/
http://www.progonos.com/
http://en.wikipedia.org/wiki/Azimuthal_equidistant_projection

References

Anderson, D.J. (1982). The home range: a new nonparametric estimation technique. *Ecology*, Vol. 63, pp. 103-112.

Avery, M.L., Humphrey, J.S., Daughtery, T.S., Fischer, J.W., Milleson, M.P., Tillman, E.A., Bruce, W.E. & Walter, W.D. (2011). Vulture flight behavior and implications for aircraft safety. *Journal of Wildlife Management*, In press

Bailey, H., Shillinger, G., Palacios, D., Bograd, S., Spotila, J., Paladino, F. & Block, B. (2008). Identifying and comparing phases of movement by 510 leatherback turtles using stateâĂŞspace models. *Journal of Experimental Marine Biology and Ecology*, Vol. 356, pp. 128-135

Benhamou, S. & Cornelis, D. (2010). Incorporating movement behavior and barriers to improve kernel home range space use estimates. *Journal of Wildlife Management*, Vol. 74, pp. 1353-1360.

Benhamou, S. (2011). Dynamic approach to space and habitat use based on biased random bridges. *PLoS ONE*, Vol. 6, e14592.

Blundell, G. M., Maier, J. A. K. & Debevec, E. M. (2001). Linear home ranges: effects of smoothing, sample size, and autocorrelation on kernel estimates. *Ecological Monographs*, Vol. 71, No.3, pp. 469-489.

Bullard, F. (1991). Estimating the home range of an animal: a Brownian bridge approach. *Thesis, The University of North Carolina*, Chapel Hill,USA.

Burt, W.H. (1943). Territoriality and home range concepts as applied to mammals. *Journal of Mammalogy*, Vol. 24, No.3, pp. 346-352.

Calenge, C. (2011) Home Range Estimation in R:, the package âĂIJadehabitatHRâĂİ for the R software, *Office national de la classe et de la faune sauvage Saint Benoist*, France.

Calenge, C. (2011) Home Range Estimation in R:, the package âĂIJadehabitatLTâĂİ for the R software, *Office national de la classe et de la faune sauvage Saint Benoist*, France.

Calhoun, J.B. & Casby, J.U. (1958). Calculation of the home range and density of small mammals, *Public Health Monograph Number 55*. United States Department of Health, Education and Welfare, Washington, D.C., USA.

Cappelle, J., Iverson, S. A., Takekawa, J. Y., Newman, S. H., Dodman, T. &

Gaidet, N. (2011). Implementing telemetry on new species in remote areas: recommendations from a large-scale satellite tracking study of African waterfowl. *Ostrich: African Journal of Ornithology*, Vol. 82, No.1, pp. 17-26.

Dixon, K.R. & Chapman, J.A. (1980). Harmonic mean measure of animal activity areas. *Ecology*, Vol. 61, pp. 1040-1044.

Don, B.A.C. & Rennolls, K. (1983). A home range model incorporating biological attraction points. *Journal of Animal Ecology*, Vol. 52, pp. 69-81.

Duong, T. & Hazelton, M. L. (2003). Plug-in bandwidth matrices for bivariate kernel density estimation. *Nonparametric Statistics*, Vol. 15, No.1, pp. 17-30.

Farmer, C., Safi, K., Barber, D.R., Martel, M. & Bildstein, K.L. (2010). Efficacy of migration counts for monitoring continental populations of raptors: an example using the osprey (Pandion haliaetus). *The Auk*, Vol. 127, pp. 863-870.

Fieberg, J. (2007). Kernel density estimators of home range: smoothing and the autocorrelation red herring. *Ecology*, Vol. 88, No.4, pp. 1059-1066.

Fieberg, J., Matthiopoulos, J., Hebblewhite, M., Boyce, M.S. & Frair, J.L. (2010). Correlation and studies of habitat selection: problem, red herring or opportunity? *Philosophical Transactions of the Royal Society B: Biological Sciences*, Vol. 365, pp. 2233-2244.

Frair, J.L., Fieberg, J., Hebblewhite, M., Cagnacci, F., DeCesare, N.J. & Pedrotti, L. (2010). Resolving issues of imprecise and habitat-biased locations in ecological analyses using GPS telemetry data. *Philosophical Transactions of the Royal Society B: Biological Sciences*, Vol. 365, pp. 2187-2200.

Gitzen, R.A., Millspaugh, J.J. & Kernohan, B.J. (2006). Bandwidth selection for fixed-kernel analysis of animal utilization distributions. *Journal of Wildlife Management*, Vol. 70, No.5, pp. 1334-1344.

Gurarie, E., Andrews, R.D.& Laidre, K.L. (2009). A novel method for identifying behavioural changes in animalmovement data. *Ecology Letters*, Vol. 12, pp. 395-408.

Hall, P., Lahiri, S.N. & Truong, Y.K. (1995). On bandwidth choice for density estimation with dependent data. *The Annals of Statistics*, Vol. 23, No.6, pp. 2241-2263.

Hemson, G., Johnson, P., South, A., Kenward, R., Ripley, R. & Mcdonald, D. (2005). Are kernels the mustard? data from global positioning system (GPS) collars suggests problems for kernel home-range analyses with least-squarescross-validation. *Journal of Animal Ecology*,Vol. 74, pp. 455-463.

Home, J. S., & Garton, E.O. (2006). Likelihood cross validation versus least squares cross-validation for choosing the smoothing parameter in kernel home-range analysis. *Journal of Wildlife Management*, Vol. 70, pp. 641-648.

Horne, J.S., Garton, E.O., Krone, S.M. & Lewis, J.S. (2007). Analyzing animal movements using brownian bridges. *Ecology*, Vol. 88, pp. 2354-2363.

Jennrich, R.I. & Turner, F.B. (1969). Measurement of non-circular home range. *Journal of Theoretical Biology*, Vol. 22, pp. 227-237.

Jones, M.C., Marron, J.S. & Sheather, S.J. (1996). A brief survey of bandwidth selection for density estimation. *Journal of the American Statistical Association*, Vol. 91, No.433, pp.401-407.

Jonsen, I.D., Flemming, J.M.& Myers, R.A. (2005). Robust stateâĂŞspacemodeling of animal movement data. *Ecology*, Vol. 86, pp. 2874-2880.

Kahle, D. & Wickham, H. ; ggmap : Spatial Visualization with ggplot2. *The R Journal*, Vol. 5/1, June.

Kie, (2013). A rule- based ad hoc method for selecting a bandwidth in kernel home-range analyses. *Animal Biotelemetry* 1-13.

Kie, J.G., Matthiopoulos, J., Fieberg, J., Powell, R.A., Cagnacci, F., Mitchell, M.S., Gaillard, J.M. & Moorcroft, P.R. (2010). The home-range concept: are traditional estimators still relevant with modern telemetry technology? *Philosophical Transactions of the Royal Society B*, Vol. 365, No.1550, pp. 2221-2231.

Kranstauber, B., Kays, R., LaPoint, S.D, Wikelski, M. & Safi, K. (2012). A dynamic Brownian bridge movement model to estimate utilization distributions for heterogeneous animal movement. *Journal of Animal Ecology* , Vol. 81, pp. 738-746.

Lewis, J.S., Rachlow, J.L. , Garton, E.O. & Vierling, L.A. (2007). Effects of habitat on GPS collar performance using data screening to reduce location error. *Journal of Applied Ecology*, Vol. 44, pp. 663-671.

Lewis, J.S., Rachlow, J.L., Horne, J.S., Garton, E.O., Wakkinen, W.L., Hayden, J. & Zager, P. (2011). Identifying habitat characteristics to predict highway crossing areas for black bears within a human-modified landscape. *Landscape and Urban Planning*, Vol. 101, pp. 99-107.

Lonergan, M., Fedak, M. & McConnell, B. (2009). The effects of interpolation error and location quality on animal track reconstruction. *Marine Mammal Science*, Vol. 25, pp. 275-282.

Lichti N. I. & Swihart R. K. (2011). Estimating utilization distribution with kernel versus local convex hull methods. *Journal of Wildlife Management*, Vol. 75, No.2, pp. 413-422.

Loader, C. R. (1999). Bandwidth selection: classical or plug-in? emphThe Annals of Statistics, Vol. 27, No.2, pp. 415-438.

Mandel, J. T., Bildstein, K. L., Bohrer, G. & Winkler, D.W. (2008). Movement ecology of migration in turkey vultures. *Proceedings of the National Academy of Sciences*, Vol. 105, No.49, pp. 19102-19107.

Morales, J.M., Haydon, D.T., Frair, J., Holsinger, K.E. & Fryxell, J.M. (2004).

Extracting more out of relocation data: building movement models as mixtures of randomwalks. *Ecology*, Vol. 85, pp. 2436-2445.

Ovaskainen, O. & Crone, E.E. (2009). Modeling animal movement with diffusion. *Spatial Ecology*. CRC Press, London,UK.

Pellerin, M., Said, S. & Gaillard, J.M. (2008). Roe deer Capreolus capreolus home-range sizes estimated from VHF and GPS data. *Wildlife Biology*, Vol. 14, No.1, pp. 101-110.

Powell, R.A. (2000). Animal home ranges and territories and home range estimators. *Research techniques in animal ecology: controversies and consequences.*, pp. 65-110, L. Boitani & T. K. Fuller editors, Columbia University Press, New York, USA.

Rodgers, A.R. & Kie, J.G. (2010). HRT: Home Range Tools for ArcGISÂő, version 1.1, 24.4.2011, Available from: `http://flash.lakeheadu.ca/~arodgers/hre/Draft%20HRT%20Users%20Manual%20Sep%2028%202010.pdf`

Ross, S.M. (1983). Stochastic processes. John Wiley, New York, USA.

Sawyer, H., Kauffman, M.J., Nielson, R.M. & Horne, J. S. (2009). Identifying and prioritizing ungulate migration routes for landscape-level conservation. *Ecological Applications*, Vol. 19, No.8, pp. 2016-2025.

Seaman, D.E. & Powell, R.A. (1996). An evaluation of the accuracy of kernel density estimators for home range analysis. *Ecology*, Vol. 77, pp. 2075-2085.

Seaman, D.E., Millspaugh, J.J., Kernohan, B.J., Brundige, G.C., Raedeke, K.J. & Gitzen R.A. (1999). Effects of sample size on kernel home range estimates. *Journal of Wildlife Management*, Vol. 63, No.2, pp. 739-747.

Silverman, B.W. (1986). Density estimation for Statistics and Data Analysis. Chapman&Hall, London, England.

Takekawa, J.Y., Newman, S.H., Xiao, X., Prosser, D.J., Spragens, K.A., Palm, E.C., Yan, B., Li, T., Lei, F., Zhao, D., Douglas, D.C., Muzaffar, S.B. & Ji, W. (2010). Migration of waterfowl in the east asian flyway and spatial relationship to HPAI H5N1 outbreaks. *Avian Diseases, 54(s1)*, pp. 466-476.

Van Winkle W. (1975). Comparison of several probabilistic home range models. *Journal of Wildlife Management*, Vol. 39, pp. 118-123.

Walter, W.D., Fischer, J.W., Baruch Mordo S. & VerCauteren, K.C. (2011). What is the proper method to delineate home range of an animal using today's advanced GPS telemetry systems: The initial step. *USDA National Wildlife Research Center*,University of Nebraska âĂŞ Lincoln.

Wand, M.P. & Jones, M.C. (1995). Kernel Smoothing. ChapmanHall, London, England.

Worton, B.J. (1987). A review of models of home range for animal movement. *Ecological Modeling*, Vol. 38, pp. 277-298.

Worton, B.J. (1989). Kernel methods for estimating the utilization distribution in home-range studies, *Ecology*,Vol. 70, pp. 164-168.

Worton, B.J. (1995). Using Monte Carlo simulation to evaluate kernel-based home range estimators. *Journal of Wildlife Management*, Vol. 59, No.4, pp. 794-800.

Willems, E.P. & Hill, R.A. (2009). Predator-specific landscapes of fear and resource distribution: effects on spatial range use. *Ecology*, Vol. 90, pp. 546-555.