

2018

Step-Selection Functions for Modeling Animal Movement -- Case Study: African Buffalo

Maia Adar

Claremont McKenna College

Recommended Citation

Adar, Maia, "Step-Selection Functions for Modeling Animal Movement -- Case Study: African Buffalo" (2018). *CMC Senior Theses*. 1938.

http://scholarship.claremont.edu/cmc_theses/1938

This Open Access Senior Thesis is brought to you by Scholarship@Claremont. It has been accepted for inclusion in this collection by an authorized administrator. For more information, please contact scholarship@cuc.claremont.edu.

Claremont McKenna College

Step-Selection Functions for Modeling
Animal Movement
Case Study: African Buffalo

submitted to

Mark Huber

by

Maia Adar

for

Senior Thesis

Spring 2018

April 23, 2018

Acknowledgements

I would like to thank Robin Naidoo of the World Wildlife Fund for mentoring me, providing me with his hard-earned data, and believing in my ability to complete this challenging project. It is a privilege to work with him and learn about his fascinating research.

I would also like to thank my thesis reader, Mark Huber, for his endless patience and consistently positive mood. He is a wonderful professor of statistics, mentor, and generally kind person to be around.

Lastly, I want to thank all the people who support me in life: my parents, Dorit and Michel Adar, for inspiring my curiosity in mathematics and my love of nature; my boyfriend, David Hurdle, for always believing in me and providing stress relief; and all of my accomplished, smart, wonderful friends for inspiring me with their energy.

Abstract

Understanding what factors influence wildlife movement allows landscape planners to make informed decisions that benefit both animals and humans. New quantitative methods, such as step-selection functions, provide valuable objective analyses of wildlife connectivity. This paper provides a framework for creating a step-selection function and demonstrates its use in a case study. The first section provides a general introduction about wildlife connectivity research. The second section explains the math behind the step-selection function using a simple example. The last section gives the results of a step-selection model for African buffalo in the Kavango Zambezi Transfrontier Conservation Area. Buffalo were found to avoid fences, rivers, and anthropogenic land use; however, there was great variation in individual buffalo's preferences.

Contents

1	Introduction	5
	Wildlife Connectivity	5
	Wildlife Corridors	5
	GPS and GIS Technology	6
	Data-Based Methods of Evaluating Connectivity	7
	Resource Selection Functions	8
	Issues in Connectivity Modeling	9
2	A Simple Example	11
	Step-Selection Function	11
	Bayesian Approach	14
3	Case Study: African Buffalo	16
	About the Data	16
	Creation of Alternative Steps	17
	Calculating Landscape Variables	18
	Estimating Coefficients	18
	Discussion	20
	Conclusion	21

1 Introduction

Wildlife Connectivity

Habitat loss is the greatest threat that most species face today [21]. Unfortunately, animals are being pushed out of their habitats by human activity, which inhibits their ability to move around and reach the resources they need to survive. The problem is more than a matter of decreasing total space – rather, the habitat becomes fragmented into smaller pieces by roads and cities, preventing key ecological processes that occur across a larger landscape. For example, habitat fragmentation greatly affects tigers because they are extremely territorial animals that require up to 400 km² of individual home range [9], so when their space is divided, the population is quickly limited. Additionally, animals trying to cross roads can be hit by traffic, and people often kill predators when they enter human areas in search of food.

In order to conserve the amazing variety of wildlife that share our planet, it is imperative to design ways for people and animals to share limited space. It is possible to make wiser decisions about land use that allow wildlife to live alongside humans, and this is the goal of connectivity research. By gaining a better understanding of what factors influence animal movement, conservation planners can make more informed decisions about land use that will benefit wildlife.

Connectivity is “the degree to which the landscape facilitates or impedes movement” [1]. In particular, functional connectivity focuses on how environmental and anthropogenic factors prevent or enable animals to move through different areas [1]. For example, since factors such as roads impede animal movement, reducing the amount of roads would increase connectivity.

Wildlife Corridors

Wildlife corridors have been used for many years to increase connectivity by facilitating wildlife movement between areas of habitat. They are essentially paths that connect larger areas of habitat that are separated by human activity, and they are crucial for preventing habitat fragmentation, allowing populations to

mix, facilitating migration, and giving animals access to more space and resources. Corridors exist at different spatial scales; some are wide expanses that connect regions and others are simply bridges or tunnels for animals to safely bypass a highway. For example, in Canada's Banff National Park, there are 24 crossing structures designed to help animals cross the Trans-Canada Highway, though research shows that they may be more useful for some types of animals than others [12].

Deciding where to place corridors so that they would actually be used by animals can be difficult. It is important to assess whether existing corridors are effective ways for animals to move between areas of habitat.

One recent study evaluated the effectiveness of both local-scale and regional-scale corridors in the Kavango Zambezi Transfrontier Conservation Area, focusing on the movements of African elephants [13]. The study used path selection functions to evaluate the connectivity of the landscape, and found that of the 9 large-scale corridors that they evaluated, 7 had high connectivity, while only 14 of 33 small-scale corridors did. Such information is valuable to conservation planners in designing corridors.

GPS and GIS Technology

Initially, the best way to understand animal movement and identify corridors was by consulting expert knowledge [1]. Local tribes, hunters, and biologists were most familiar with animal movement patterns, and their opinion was the best information available about where animals go and which factors affect their movements. Today, data-based models are preferred for objective evaluation of connectivity.

Advances in geospatial tracking technology have allowed for new developments in connectivity research in recent years. The Global Position System (GPS) became fully operational in 1995 when the U.S. government launched the 24th satellite into orbit, completing a network of satellites that allows for precise determination of location anywhere on the globe [15]. GPS computes the three-dimensional position of a point by timing how long it takes to send a signal from

a satellite to the Earth, and then calculating the distance between the point on Earth and four different satellites.

GPS technology has allowed wildlife biologists to gather information about animal movements that was previously inaccessible. By putting tracking devices on animals, it is now possible to know animals' locations at all times, leading to great advances in understanding their behavior. There are some limitations to GPS, since it can be expensive, difficult to attach to an animal, and less accurate under forest canopies or underground [18]. However, technology is continuously improving and mitigating some of these issues.

GPS data can be studied in comparison to layers within a geographical information system (GIS). In addition to an animal's position, a GIS reveals details about the environment at that location, allowing for inference about the animal's habitat preferences. GIS works with two main data types: vectors and rasters. Vectors can be points, lines, or polygons, representing objects at particular locations (ex. animal locations, roads, lakes). Rasters represent a continuous surface using a grid with cells that each hold a value (ex. temperature, elevation, land-cover categories).

Data-Based Methods of Evaluating Connectivity

The first step in evaluating connectivity is creating a resistance layer. Resistance, as defined by Zeller et al. [24], is “the willingness of an organism to cross a particular environment, the physiological cost of moving through a particular environment, the reduction in survival for the organism moving through a particular environment, or an integration of all these factors.” The resistance layer assigns a resistance value for each cell in a GIS raster, representing how costly it is for the animal to move there. These values are determined by a statistical model with various environmental variables as the predictors.

The environmental variables must be wisely chosen to be relevant to the particular animal's behavior. Additionally, one must consider the effect of scale in representing these variables. There are often limitations in gathering these data; for example, most spatial data is often collected via remote sensing, which can

be inaccurate [7].

A resource selection function (RSF) produces estimates of resistance using data about animals' GPS location and data about environmental variables. There are different variations of RSFs, including point selection functions, step-selection functions (SSF), and path-selection functions (PSF) [24]. They all have a similar theoretical basis, and they are covered in more detail in the next section.

After creating the resistance surface, there are several ways to find and evaluate corridors. The simplest way is the least-cost path, which finds the paths of least resistance [20]. Resistant kernel modeling is similar, but adds a dispersal function to account for the random movements of animals [3, 10]. Alternatively, one can use circuit theory [13], graph theory, or network flow to model how movements might pass through the resistance layer. This paper will focus on the creation of the resistance layer, but for more details on further steps in connectivity modeling, see the Center for Large Landscape Conservation's guide [1] or the USDA's guide [20].

Resource Selection Functions

A resource selection function weights the probability of selecting a particular unit:

$$w(\vec{x}) = \exp(\beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p)$$

where \vec{x} is the vector of predictor covariates measured in the landscape, and β are the coefficients representing the impact of each variable [11]. Units with higher $w(\vec{x})$ are more likely to be used.

The RSF compares 'used' vs. 'available' units, which can be locations, steps, or paths. The betas coefficients are estimated using maximum likelihood estimation, and the probability that unit \vec{x} will be used is

$$P(\vec{x} \text{ is used}) = \frac{w(\vec{x})}{1 + w(\vec{x})}.$$

The step-selection function (SSF) is a variation on RSF which was first used by Fortin et al. [8] for a study about elk movement. Rather than comparing units such as locations, they used the elk's "step" as the sampling unit. They measured

the elk's location every 5 hours using GPS collars, and the distance between two consecutive locations was considered a step. Instead of measuring variables at the end location, as one would do in an RSF, they measured them along the step. The available steps are taken from a distribution around the starting point with random lengths and angles.

Another variation on the RSF is the Path-Selection Function. In this case, the entire path of an animal, which includes multiple "steps," is rotated from the starting point. This method was used by Naidoo et al. [13].

Issues in Connectivity Modeling

Choosing an appropriate scale

Appropriately selecting both a time scale and a geographic scale is critical. It will depend on the particular animal's biology and the available data. GPS collar acquisition interval can vary from minutes to days. Models have been found to be sensitive to both temporal scale and GPS acquisition interval, leading to different home range estimates, ranges of resistance values, and regression coefficients [23]. Thus, it is important to consider different scales; Zeller et al. [23] even developed a method using multi-scale modeling. This can be done by creating multiple models with different scales and selecting the one that has the best result by some evaluation criteria, such as AIC. Ideally, the smaller GPS collar acquisition interval, the more flexibility there is in choosing a scale.

Geographic scale will depend on the species and the type of movement being modeled. For example, studying the daily movements of crocodiles would require a finer scale than the long migrations of the African elephant.

The Number of Available Steps

How many available alternatives should be used? And how should these random options be selected? Using fewer random alternatives makes computations more efficient, and a low number of 'available' options usually suffice [19], but sometimes more are necessary to get accurate coefficient estimates. One way of selecting steps is by randomly selecting a step length and angle from two indepen-

dent distributions. However, in some cases, step length and angle are correlated; for example, a puma may be turning around a small area while hunting, and then taking longer steps as it moves far in one direction [19]. Another method is to use a density kernel around the step or path, such as a Pareto distribution, considering the proportion of different landscape features within the kernel as the predictor covariates of the ‘available’ [23].

Measuring Landscape Variables

Another issue is deciding how to measure the landscape variables. For a step-selection function, variables can be measured at the endpoint, at the maximum value [8], as an average along the step [8, 13], or within a buffer around the step [2, 5]. Note that an animal might not actually travel in a straight line from one point to the next, but the SSF assumes that the step selection is related to the environmental variables along the step. It is a reasonable assumption because spatial autocorrelation of landscape variables causes nearby areas to have similar characteristics.

Additionally, distance to nearest feature could be a variable [2]. For example, if measuring the effect of roads on an animal’s movements, the measurement could be the distance to the nearest road, or an indicator variable indicating if the step crosses a road. Note that using the distance to nearest feature assumes that the animal has knowledge of the surrounding landscape.

Choosing which measurement method to use can be trivial in some cases, but significant in others. Thurfjell et al. gives the example of a wild boar foraging in the edge of a crop field compared to one in the middle of the crop field [19]. The step-selection function would suggest that the boar on the edge avoids the forest more than the one in the middle, because its random steps would be likely to go into the forest, even if in reality they are equally likely to avoid forest.

2 A Simple Example

Step-Selection Function

Suppose we are studying one antelope moving through a landscape which is described by the amount of grass, water, and trees in each area. We would like to know how much each variable matters to the antelope as he chooses where to move. In order to create this model, we will compare the locations that the antelope has visited ('used') to other potential locations that he did not visit ('available').

We record his location with a GPS tracking collar and find that he travels from S_1 through S_5 . On step S_2 , the used location was Y_3 , but he could have visited three alternative locations Y_1 , Y_2 , or Y_4 .

Figure 1: Example of Antelope's Path

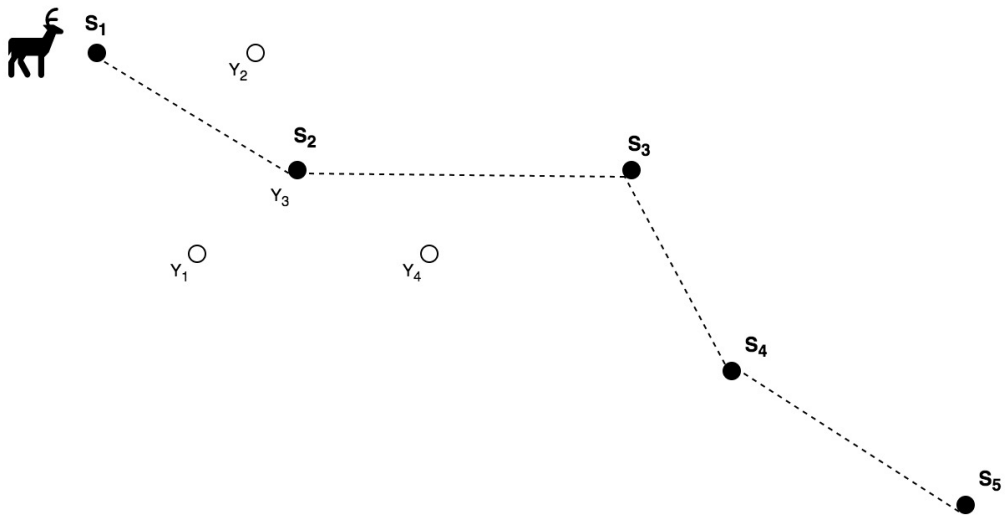


Table 1: Environmental Variables Step S_2

	$Y_{2,1}$	$Y_{2,2}$	$Y_{2,3}$	$Y_{2,4}$
Grass	2	10	2	6
Water	4	7	5	5
Trees	1	1	8	3

Now suppose we have collected data on the amount of grass, water, and trees along each possible step. We would like to find the coefficients of β which represent how much the antelope cares about each variable. To avoid confusion when we add more steps later on, let the first number in the subscript indicate the step and the second number indicate the location. So $Y_{1,3}$ is the third available option in Step S_1 . Table 1 gives the environmental variables in each location for step S_2 .

The utility function for each location is given by

$$w(x_{i,j}) = \exp(\beta_1 x_{i,1} + \beta_2 x_{i,2} + \cdots + \beta_j x_{i,j}),$$

where $x_{i,j}$ is the environmental data describing location $Y_{i,j}$. So if we suppose that $\beta = (0.3, 0.3, 0.2)$, then $w(x_{2,3}) = 54.6$.

The probability of choosing location $Y_{i,j}$, given that the animal can only choose one location in each step, is calculated by the following formula:

$$P(Y_{i,j} = 1 | Y_{i,1} + \cdots + Y_{i,j} = 1) = \frac{w(x_{i,j})}{w(x_{i,1}) + \cdots + w(x_{i,j})} = \frac{e^{\beta^T x_{i,j}}}{e^{\beta^T x_{i,1}} + \cdots + e^{\beta^T x_{i,j}}}.$$

So according to our model, the probability of choosing location $Y_{2,3}$ (which was the actual chosen location) is $P(Y_{2,3} = 1 | Y_{2,1} + Y_{2,2} + Y_{2,3} + Y_{2,4} = 1) = 0.26$. However, location $Y_{2,2}$ has probability 0.52, which is higher than the probability of choosing the used location. We might need to adjust our β coefficients.

Let us extend the example to include the antelope's second step. The data for the second step is given in Table 2, and this time suppose location $Y_{3,1}$ is used.

The likelihood function is the product of the probabilities of choosing each step. For the first step,

Table 2: Environmental Variables Step S_3

	$Y_{3,1}$	$Y_{3,2}$	$Y_{3,3}$	$Y_{3,4}$
Grass	10	3	9	4
Water	2	2	5	4
Trees	1	2	4	3

$$P(Y_{2,3} = 1 | Y_{2,1} + Y_{2,2} + Y_{2,3} + Y_{2,4} = 1) = \frac{e^{\beta^T x_{2,3}}}{\sum_{i=1}^4 e^{\beta^T x_{2,i}}},$$

and for the second step,

$$P(Y_{3,1} = 1 | Y_{3,1} + Y_{3,2} + Y_{3,3} + Y_{3,4} = 1) = \frac{e^{\beta^T x_{3,1}}}{\sum_{i=1}^4 e^{\beta^T x_{3,i}}}.$$

The likelihood function is the product of these probabilities:

$$L(\beta | x_1, \dots, x_n) = \frac{e^{\beta^T x_{2,3}}}{\sum_{i=1}^4 e^{\beta^T x_{2,i}}} \cdot \frac{e^{\beta^T x_{3,1}}}{\sum_{i=1}^4 e^{\beta^T x_{3,i}}},$$

so the log-likelihood function is

$$\ln(L(\beta | x_1, \dots, x_n)) = \ln \left(\frac{e^{\beta^T x_{2,3}}}{\sum_{i=1}^4 e^{\beta^T x_{2,i}}} \cdot \frac{e^{\beta^T x_{3,1}}}{\sum_{i=1}^4 e^{\beta^T x_{3,i}}} \right).$$

We must find the β coefficients that maximize the log-likelihood.

$$\begin{aligned} \ln(L(\beta | x_1, \dots, x_n)) &= \ln \left(\frac{e^{\beta^T x_{2,3}}}{\sum_{i=1}^4 e^{\beta^T x_{2,i}}} \right) + \ln \left(\frac{e^{\beta^T x_{3,1}}}{\sum_{i=1}^4 e^{\beta^T x_{3,i}}} \right) \\ &= \ln(e^{\beta^T x_{2,3}}) + \ln(e^{\beta^T x_{3,1}}) - \ln(\sum_{i=1}^4 e^{\beta^T x_{2,i}}) - \ln(\sum_{i=1}^4 e^{\beta^T x_{3,i}}). \\ &= \beta^T (x_{2,3} + x_{3,1}) - \ln(\sum_{i=1}^4 e^{\beta^T x_{2,i}}) - \ln(\sum_{i=1}^4 e^{\beta^T x_{3,i}}). \end{aligned}$$

This is the furthest the log-likelihood can be easily simplified. To maximize it, we can use the mle function in R.

```
1 nLL <- function (b1, b2, b3) {
2   B <- as.vector(c(b1, b2, b3))
```

```

3  x <- matrix(c(2,10,2,6,4,7,5,5,1,1,8,3), nrow=4, ncol=3)
      #data from Table 1
4  x2 <- matrix(c(10,3,9,4,2,2,5,4,1,2,4,3), nrow=4, ncol=3)
      #data from Table 2
5
6  LL <- (B %*% x[3,]) + (B %*% x2[1,]) - log(exp(B %*% x
      [1,]) + exp(B %*% x[2,]) + exp(B %*% x[3,]) + exp(B %*
      % x[4,])) -
7  log(exp(B %*% x2[1,]) + exp(B %*% x2[2,]) + exp(B %*%
      x2[3,]) + exp(B %*% x2[4,])) #log likelihood
8  return (-LL)
9 }
10
11 mle(nLL, start=list(b1 = 0, b2 = 0, b3 = 0), method="BFGS")

```

This gives $\beta_1 = 1.9$, $\beta_2 = -6.2$, and $\beta_3 = 3.6$. With these β estimates, the probability of choosing $Y_{2,3} \approx 1$, and the probability of choosing the others is 0. Similarly, in the next step the probability of choosing $Y_{3,1} \approx 1$ and the other locations are 0. This is because the model only has information about two steps, so it is easy to fit the β coefficients perfectly.

When we extend the model to six steps using randomly generated data, we get $\beta_1 = 0.21$, $\beta_2 = -0.22$, and $\beta_3 = -0.0058$. For the second step, these estimates give $P(Y_{3,1}) = 0.57$, $P(Y_{3,2}) = 0.10$, $P(Y_{3,3}) = 0.23$, and $P(Y_{3,4}) = 0.10$. The chosen step $Y_{3,1}$ still has the highest probability of being chosen according to the model, but the prediction is less certain.

Bayesian Approach

Another option is to take a Bayesian approach. Suppose the prior distribution for each β is the standard Cauchy distribution, with density

$$f(b) = \frac{1}{\pi(1+b^2)}.$$

The posterior distribution is proportional to the prior times the likelihood.

$$\text{posterior} \propto \frac{1}{1 + \beta_1^2} \cdot \frac{1}{1 + \beta_2^2} \cdot \frac{1}{1 + \beta_3^2} \cdot L(\beta|x_1, \dots, x_n)$$

To find the posterior mode, simply maximize this function. Using the same likelihood as before, with steps S2 and S3, we must find

$$\arg \max_{\beta_1, \beta_2, \beta_3} \frac{1}{1 + \beta_1^2} \cdot \frac{1}{1 + \beta_2^2} \cdot \frac{1}{1 + \beta_3^2} \cdot \frac{e^{\beta^T x_{2,3}}}{\sum_{i=1}^4 e^{\beta^T x_{2,i}}} \cdot \frac{e^{\beta^T x_{3,1}}}{\sum_{i=1}^4 e^{\beta^T x_{3,i}}}.$$

This gives $\beta_1 = 0.25$, $\beta_2 = -0.65$, and $\beta_3 = 0.45$. Note that the previous estimates were different ($\beta_1 = 1.9$, $\beta_2 = -6.2$, and $\beta_3 = 3.6$). However, we still get a high probability of choosing the chosen step, with $P(Y_{2,3}) = 0.93$. The β coefficients are not as perfectly fit as before because we are now including the information about the prior distribution, in addition to the information about the two steps.

3 Case Study: African Buffalo

About the Data

Established in 2011, the Kavango Zambezi Transfronteir Conservation Area spans across Angola, Botswana, Namibia, Zambia, and Zimbabwe and totals 106 million acres of land [22]. The goal of this international project is to “sustainably manage the Kavango Zambezi ecosystem, its heritage and cultural resources based on best conservation and tourism models for the socio-economic wellbeing of the communities and other stakeholders in and around the eco-region through harmonization of policies, strategies and practices” (<http://www.kavangozambezi.org/index.php/en/>) [17]. Wildlife connectivity research can provide policy-makers with the information needed to meet these goals, but not many studies have researched wildlife connectivity in this area. Past studies have focused on lions [4, 6] and elephants [3, 13], but not African buffalo.

For this study, the GPS location data of 42 African buffalo in the Namibian part of KAZA were collected across a 10 year period. Each animal’s location was recorded every 5 hours. The buffalo came from different herds, so their movements represent the movements of many buffalo. In total there were 105,799 locations recorded between September 2007 and November 2017. However, due to errors in the GPS, only 63,402 5-hour steps were recorded, leaving data on 36 buffalo.

The following environmental data were used as predictor variables for the model:

- Anthropogenic Factors
 - Rivers
 - Roads
 - Fence
 - Anthropogenic Land Use

- Natural Factors
 - Waterholes
 - Treecover

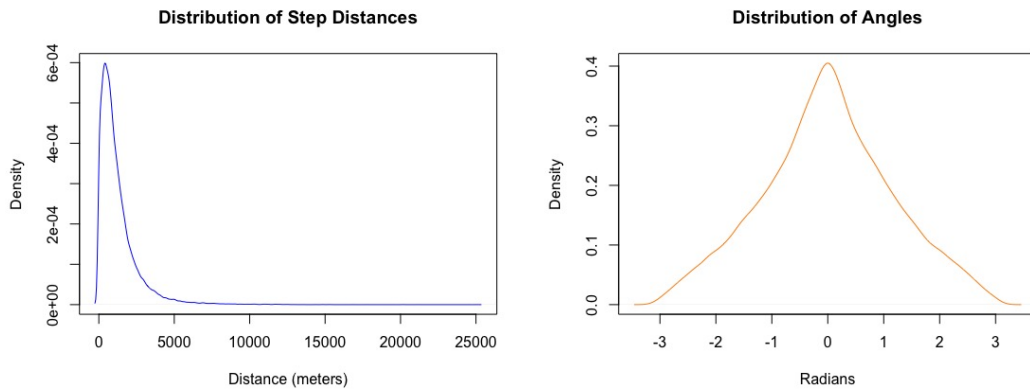
- Woodland
- Grassland
- Bushland
- Floodplain

Location of rivers, roads, and fences were taken from regional GIS databases. Waterhole locations were compiled by Robin Naidoo from Sentinel satellite imagery. The treecover data came from the MOD44B dataset which measures global surface vegetation. The other landcover data (anthropogenic land use, woodland, floodplain, grassland, bushland) was taken from satellite imagery collected by The Landsat Program in 2005.

Creation of Alternative Steps

Alternative steps were created using a random distance and turning angle. The distance was chosen from the distribution of 5-hour step distances of all buffalo recorded in the study; similarly, the turning angles were chosen from the distribution of all turning angles [8].

Figure 2: Distributions of Step Distance and Angles



The buffalos tend to move about 1 km every 5 hours, and usually not more than 5 km, but they occasionally take much longer steps. The longest recorded step was 25.1 km in 5 hours.

As we would expect, they tend to continue traveling in the same direction, so the peak of the turning angle distribution is near 0. For calculation of turning

angles, see Appendix A.

Three alternative steps were created for each step. Although Northrup et al. [14] found that the coefficient estimates do not converge until a certain number of random alternative steps are used, Thurfjell et al. [19] suggest that this is not a problem for SSFs since they usually have a large sample size, and that few or even one alternative step is enough.

Calculating Variables

Roads, rivers, fences, and waterholes were line features which were coded as indicator variables (“1” if crossed or “0” if not). Buffalo tended not to cross these features often. In total, rivers were crossed 4,137 times, fences were crossed 1,093 times, and roads were crossed 5,708 times. Waterholes were polygon features which were also coded as indicator variables. Waterholes were crossed a total of 30,972 times. Note that “crossed” does not necessarily mean the animal crossed that feature, but that the line between two consecutive GPS measurement points crossed the feature.

Treecover was measured as an average value along the step. Woodland, floodplain, bushland, grassland, and anthropogenic land use were measured as the proportion of the step which covered that land type.

See Appendix C for R code demonstrating how variable calculations were made.

Estimating Coefficients

The methodology for estimating coefficients followed that of Squires et al. [16]. A model was created for each individual buffalo using conditional logistic regression, similar to the method that was described in Section 2. For details on how these calculations were implemented in R, see Appendix D.

Then, the coefficients were averaged across the individual models. A t-test was used to determine if each coefficient was different from 0 with a 5% chance of Type I error. Figure 3 shows the distribution of beta estimates across individuals for each variable.

Figure 3: Distribution of Coefficients for Each Variable

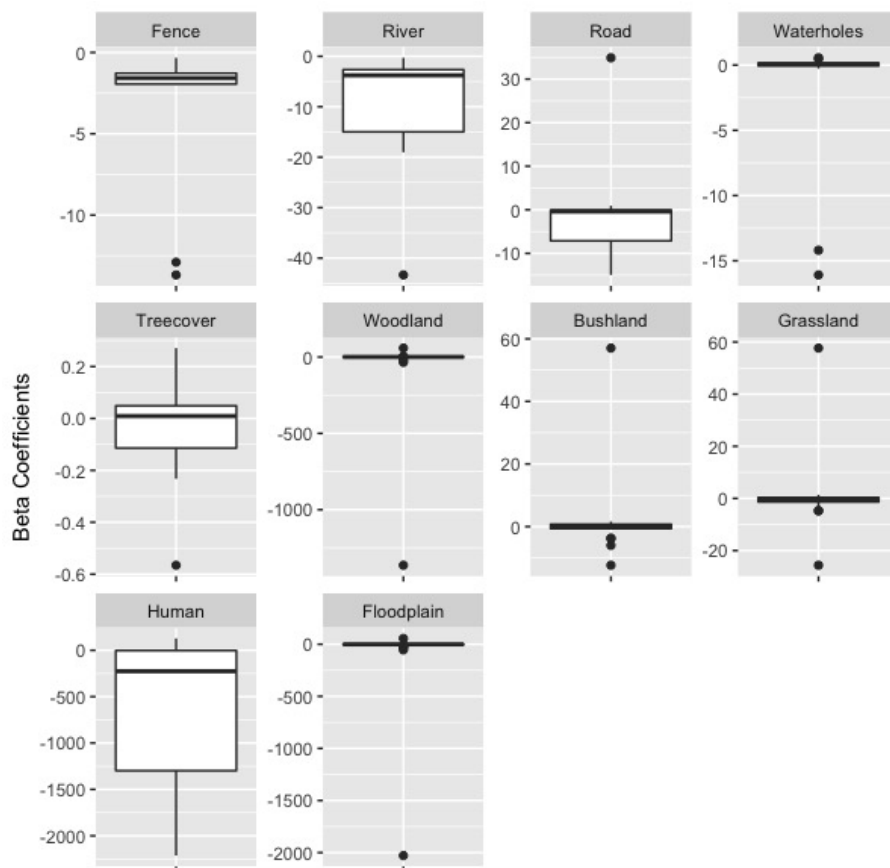


Table 3: Coefficient estimates for population-level model

Variable	Estimate	Standard Error	p-value
Fence	-1.27	0.130	< 0.0001
River	-3.06	0.113	< 0.0001
Human	-2.10	0.250	< 0.0001

Only Fence, River, and Human were statistically significant at the 0.05-level, so these three variables were used in the final population-level model summarized in Table 3.

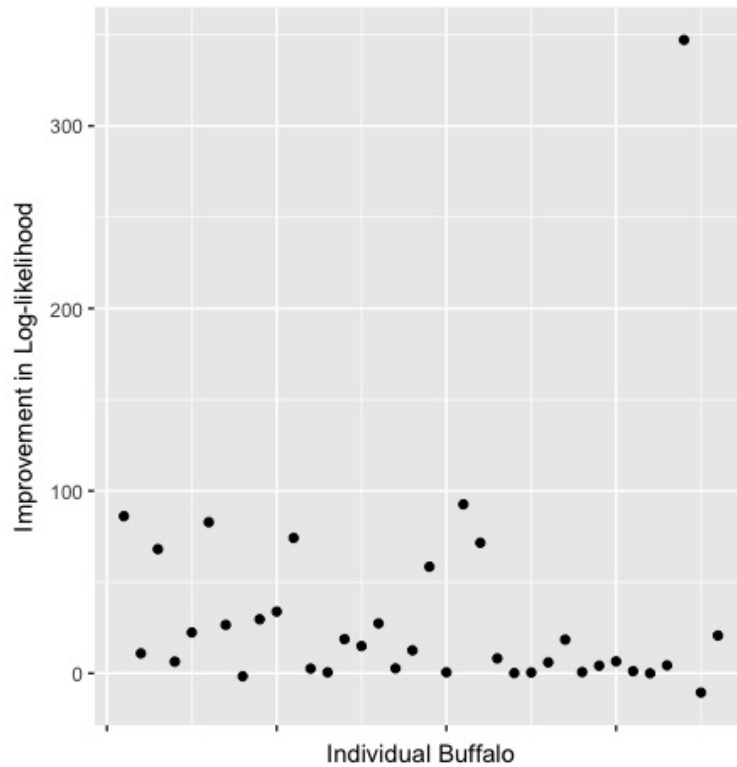
In order to determine how well the population-level model performed for individual buffalo, log-likelihood using the population-level model coefficients was compared to log-likelihood using 0 for all coefficients for each individual buffalo. A greater log-likelihood indicates better fit, so a positive difference between the population-level model and the null model indicates that the model is a good fit for that individual.

Discussion

In the distributions of beta coefficients (Figure 3), Fence, River, and Human (anthropogenic land use) visibly differ from 0, while most of the other distributions overlap 0. These three variables were significant at the 5% level. Road also appears to be exert a negative influence on buffalo movement, but the difference was not great enough to be significant, possibly due to the single large outlier. It should be noted that many of the coefficient distributions had large outliers, suggesting that there is great variation in individual preferences. Also, the coefficients for Human were very large compared to the other variables, with a mean value of -648 and a minimum of -2213, suggesting that most buffalo are strongly repelled by anthropogenic land use.

All three coefficients in the population-level model are negative, meaning that buffalo are deterred by fences, rivers, and anthropogenic land use.

Figure 4: Improvement in Log-likelihood Using the Population-Level Model



The difference in log-likelihood between the population-level model and the null model indicates how well the population-level model fit for each individual buffalo. Two buffalo had a slightly negative difference, signaling that the model was not at all suitable for those buffalo. Although the model performed better than the null for the other buffalo, most improvements were modest. This implies great variation between individual preferences, which makes it difficult to create a model at the population-level. A hierarchical model with individuals as random effects would likely better represent the buffalo’s individual differences.

As discussed in the introduction, one of the big questions in step-selection function modeling is choosing a suitable number of alternative steps. Although this study used three alternatives based on the recommendation of [19], this decision lacked the evidence to be made confidently, and repeating the study with different numbers of alternatives could produce valuable results for step-selection modeling research.

Conclusion

Overall, it is not surprising that fences, rivers, and anthropogenic land use exerted a negative effect on step-selection. However, more analysis is needed to understand the magnitude of each variable's effect, and to better model individual differences between buffalo. In particular, a hierarchical model with individuals as random effects would be a better model for this data. Next steps include creation of a resistance layer, estimation of connectivity, and evaluation of existing wildlife corridors.

This paper provides a framework for creating a step-selection function. Section 2 explains the math behind the modeling, and the appendices provide detailed, reproducible code for cleaning and analyzing data. Additionally, it develops methods for calculating the distribution of step distances and turning angles, and evaluating a population-level model for individual buffalo. It is also the first paper to analyze this data about buffalo in KAZA.

Step-selection models are an effective way of quantifying animal movement to objectively analyze connectivity, which is valuable for planning conservation areas such as the Kavango Zambezi Transfronteir Conservation Area. The results from modeling buffalo should be compared to results of other animals in KAZA for a fuller understanding of connectivity across species. These types of studies provide powerful insight about strategies for planning land use in order to best benefit humans and wildlife.

References

- [1] R. Ament, R. Callahan, M. McClure, M. Reuling, and G. Tabor. *Wildlife Connectivity: Fundamentals for conservation action*. Center for Large Landscape Conservation, 2014.
- [2] A. Coulon, N. Morellet, M. Goulard, B. Cargnelutti, J. Angibault, and A. J. M. Hewison. Inferring the effects of landscape structure on roe deer (*capreolus capreolus*) movements using a step selection function. *Landscape Ecology*, 23:603–614, 2008.
- [3] S. A. Cushman, M. Chase, and C. Griffin. Mapping landscape resistance to identify corridors and barriers for elephant movement in southern africa. In S. A. Cushman and F. Huettmann, editors, *Spatial Complexity, Informatics, and Wildlife Conservation*, pages 349–367. Springer, 2010.
- [4] S. A. Cushman, N. B. Elliot, D. W. Macdonald, and A. J. Loveridge. A multi-scale assessment of population connectivity in African Lions (*panthera leo*) in response to landscape change. *Landscape Ecology*, 31:1337–1353, 2016.
- [5] B. G. Dickson, J. S. Jenness, and P. Beier. Influence of vegetation, topography, and roads on cougar movement in southern california. *The Journal of Wildlife Management*, 69:264–276, 2005.
- [6] N. B. Elliot, S. A. Cushman, D. W. Macdonald, and A. J. Loveridge. The devil is in the dispersers: predictions of landscape connectivity change with demography. *Journal of Applied Ecology*, 51:1169–1178, 2014.
- [7] G. M. Foody. Status of land cover classification accuracy assessment. *Remote Sensing of Environment*, 80:185–201, 2002.
- [8] D. Fortin, H. L. Beyer, M. S. Boyce, D. W. Smith, T. Duchesne, and J. S. Mao. Wolves influence elk movements: Behavior shapes a trophic cascade in yellowstone national park. *Ecology*, 86:1320–1330, 2005.
- [9] J. Goodrich, A. Lynam, D. Miquelle, H. Wibisono, K. Kawanishi, A. Patanavibool, S. Htun, T. Tempa, J. Kark, Y. Jhala, and U. Karanth. *Panthera*

- tigris. Retrieved 10 April 2018, <http://www.iucnredlist.org/details/15955/0>.
- [10] R. Krishnamurthy, S. A. Cushman, M. S. Sarkar, M. Malviya, M. Naveen, J. A. Johnson, and S. Sen. Multi-scale prediction of landscape resistance for tiger dispersal in central india. *Landscape Ecology*, 31:1355–1368, 2016.
- [11] B. F. J. Manly, L. L. McDonald, D. L. Thomas, T. L. McDonald, and W. P. Erickson. *Resource Selection by Animals*. Kluwer Academic Publishers, 2002.
- [12] W. McDonald and C. C. St Clair. Elements that promote highway crossing structure use by small mammals in banff national park. *Journal of Applied Ecology*, 41:82–93, 2004.
- [13] R. Naidoo, J. W. Kilian, P. Du Preez, P. Beytell, O. Aschenborn, R. D. Taylor, and G. Stuart-Hill. Evaluating the effectiveness of local- and regional-scale wildlife corridors using quantitative metrics of functional connectivity. *Biological Conservation*, 217:96–103, 2018.
- [14] J. M. Northrup, M. B. Hooten, C. R. Anderson, and G. Wittemyer. Practical guidance on characterizing availability in resource selection functions under a useavailability design. *Ecology*, 94:1456–1463, 2013.
- [15] S. Pace, G. P. Frost, I. Lachow, D. R. Frelinger, D. Fossum, D. Wassem, and M. M. Pinto. *The Global Positioning System: Assessing National Policies*. RAND Corporation, 1995.
- [16] J. R. Squires, N. J. DeCesare, L. E. Olson, J. A. Kolbe, M. Hebblewhite, and S. A. Parks. Combining resource selection and movement behavior to predict corridors for canada lynx at their southern range periphery. *Biological Conservation*, 157:187–195, 2013.
- [17] KAZA TFCA. About KAZA. Retrieved 10 April 2018, <http://www.kavangozambezi.org/index.php/en/about/about-kaza>.
- [18] B. Thomas, J. D. Holland, and E. O. Minot. Wildlife tracking technology options and cost considerations. *Wildlife Research*, 38:653–663, 2012.

- [19] H. Thurfjell, S. Ciuti, and M. S. Boyce. Applications of step-selection functions in ecology and conservation. *Movement Ecology*, 2, 2014.
- [20] A. A. Wade, K. S. McKelvey, and M. K. Schwartz. *Resistance-Surface-Based Wildlife Conservation Connectivity Modeling*. U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station, 2015.
- [21] WWF. Impact of habitat loss of species. Retrieved 10 April 2018, http://wwf.panda.org/about_our_earth/species/problems/habitat_loss_degradation/.
- [22] WWF. What kinds of animals live in KAZA? and four other KAZA facts. Retrieved 10 April 2018, <https://www.worldwildlife.org/stories/what-kind-of-animals-live-in-kaza-and-four-other-kaza-facts>.
- [23] K. A. Zeller, K. McGarigal, S. A. Cushman, P. Beier, T. W. Vickers, and W. M. Boyce. Using step and path selection functions for estimating resistance to movement: pumas as a case study. *Landscape Ecology*, 31:1319–1335, 2016.
- [24] K. A. Zeller, K. McGarigal, and A. R. Whiteley. Estimating landscape resistance to movement: a review. *Landscape Ecology*, 27:777–797, 2012.

Appendix A

The original data consisted of the X-Y locations of each buffalo at each time. The data was sorted and split into separate dataframes for each individual buffalo. I added the ending X-Y location for each step (which was simply the X-Y of the following step) because ArcGIS can easily translate this data into line objects using the start and end points. Then I calculated the time difference for each step and removed any that were not 5 hours. I also removed steps that were 0 length. These dataframes represented the “used” steps of the buffalo.

Next, I created the alternative steps. I calculated the distance of each step using the start and end points. I calculated the angle of each step using some trigonometry. I then randomly selected a distance and angle from the distributions of distances and angles for each alternative step. I repeated three times to create three “alternative” step files for each buffalo.

```
1 ### Creating Used Step Files for Each Buffalo ###
2
3 setwd("/Users/maia/Desktop/Thesis")
4 GPS <- read.csv("GPS.csv", header=T)
5 GPS$Date.time <- chron(substr(as.character(GPS$Date.time)
6     ,1,10), substring(as.character(GPS$Date.time),12),format
7     =c(dates="y-m-d",times="h:m:s"))
8 GPS$ID <- sub("^", "B", GPS$ID )
9
10
11 # Sort by date within ID group
12 GPS <- GPS[with(GPS,order(ID,Date.time)),]
13
14 # Split into individual buffalo
15 GPS_bybuff <- split(GPS, GPS$ID)
16
17 # Starting and ending points
18 for (b in 1:length(GPS_bybuff)){
19     names(GPS_bybuff[[b]])[names(GPS_bybuff[[b]]) == 'X'] <-
20         'startX'
```

```

17 names(GPS_bybuff[[b]])[names(GPS_bybuff[[b]]) == 'Y'] <-
    'startY'
18 GPS_bybuff[[b]]$endX <- c(GPS_bybuff[[b]]$startX[-1], NA)
19 GPS_bybuff[[b]]$endY <- c(GPS_bybuff[[b]]$startY[-1], NA)
20 }
21
22 # Calculate time difference
23 for (b in 1:length(GPS_bybuff)){
24   timediff <- diff(GPS_bybuff[[b]]$Date.time)
25   timediff <- c(as.numeric(timediff)*24, NA)
26   GPS_bybuff[[b]]$timediff <- round(as.numeric(timediff))
27 }
28
29 # Remove non-5hour steps
30 for (b in 1:length(GPS_bybuff)){
31   GPS_bybuff[[b]] <-GPS_bybuff[[b]][!(GPS_bybuff[[b]]$
    timediff!=5),]
32 }
33
34 # Remove 0 length steps
35 for (b in 1:length(GPS_bybuff)){
36   GPS_bybuff[[b]] <- GPS_bybuff[[b]][!(GPS_bybuff[[b]]$
    startX == GPS_bybuff[[b]]$endX & GPS_bybuff[[b]]$
    startY == GPS_bybuff[[b]]$endY),]
37 }
38
39 # Remove NA step that gets tacked onto the end for some
    reason
40 for (b in 1:length(GPS_bybuff)){
41   GPS_bybuff[[b]] <- na.omit(GPS_bybuff[[b]])
42 }
43
44 # Write separate buffalo files

```

```

45 setwd("/Users/maia/Desktop/Thesis/individual")
46 lapply(1:length(GPS_bybuff), function(i) write.csv(GPS_
    bybuff[[i]], file = paste0(names(GPS_bybuff[i]), ".csv")
    , row.names = FALSE))
47
48
49
50 ### Calculating Distance and Turning Angle Distributions
    ###
51
52 # Calculate distances distribution
53 distances <- vector()
54 for (b in 1:length(GPS_bybuff)) {
55   for (i in 1:nrow(GPS_bybuff[[b]])) {
56     dx <- GPS_bybuff[[b]]$startX[i] - GPS_bybuff[[b]]$endX[
        i]
57     dy <- GPS_bybuff[[b]]$startY[i] - GPS_bybuff[[b]]$endY[
        i]
58     distances <- c(distances, sqrt(dx^2 + dy^2))
59   }
60 }
61
62 distances <- na.omit(distances)
63
64 plot(density(distances), main = "Distribution of Step
    Distances", xlab = "Distance (meters)",
65       col="blue")
66
67 # Calculate turning angles distribution
68 angles <- vector()
69 for (b in 1:length(GPS_bybuff)) {
70   for (i in 1:nrow(GPS_bybuff[[b]])-1) {
71     dx1 <- GPS_bybuff[[b]]$endX[i] - GPS_bybuff[[b]]$startX

```

```

      [i]
72   dy1 <- GPS_bybuff[[b]]$endY[i] - GPS_bybuff[[b]]$startY
      [i]
73   dx2 <- GPS_bybuff[[b]]$endX[i+1] - GPS_bybuff[[b]]$
      startX[i+1]
74   dy2 <- GPS_bybuff[[b]]$endY[i+1] - GPS_bybuff[[b]]$
      startY[i+1]
75
76   if(min(length(dx1), length(dx2), length(dy1), length(
      dy2))>0){
77     if(dx1!=0) theta1 <- atan(dy1/dx1)
78     else if(dy1>0)
79       theta1 <- pi/2
80     else if(dy1<0)
81       theta1 <- -pi/2
82     else
83       theta1 <- 0
84
85
86     if(dx2!=0)
87       theta2 <- atan(dy2/dx2)
88     else if(dy2>0)
89       theta2 <- pi/2
90     else if(dy2<0)
91       theta2 <- -pi/2
92     else
93       theta2 <- 0
94
95     angles <- c(angles, theta1 - theta2)
96   }
97 }
98 }
99 angles <- na.omit(angles)

```

```

100
101 plot(density(angles), main = "Distribution of Angles", xlab
      = "Radians",
102       col="darkorange")
103
104 max(angles)
105
106
107
108 ### Creating Alternative Step Files for each Buffalo ###
109
110 # Change Step
111 for (b in c(1:34, 41:42)) {
112   for (i in 1:nrow(GPS_bybuff[[b]])){
113     radius <- distances[sample.int(length(distances), 1)]
114     angle <- angles[sample.int(length(angles), 1)]
115     dx <- cos(angle)*radius
116     dy <- sin(angle)*radius
117     GPS_bybuff[[b]]$endX[i] <- GPS_bybuff[[b]]$startX[i] +
      dx
118     GPS_bybuff[[b]]$endY[i] <- GPS_bybuff[[b]]$startY[i] +
      dy
119   }
120 }
121
122
123 # Write alternative step files
124 # For each new alternative step, must run code from the
      beginning, then write new alternative step files
125
126 setwd("/Users/maia/Desktop/Thesis/individual/alt1")
127 lapply(1:length(GPS_bybuff), function(i) write.csv(GPS_
      bybuff[[i]], file = paste0(names(GPS_bybuff[i]), "alt1.

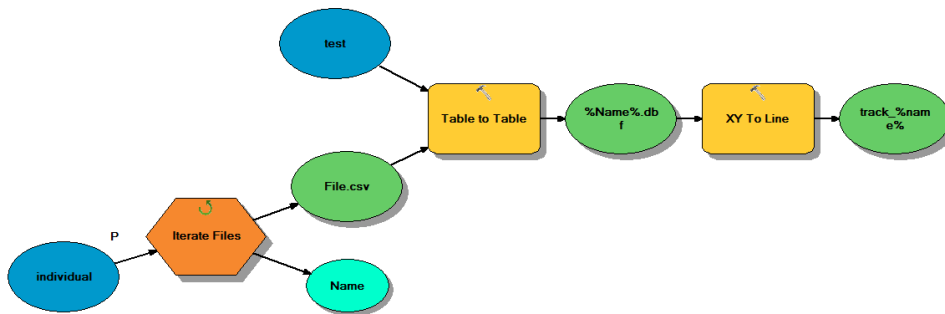
```



```
    csv"), row.names = FALSE))
128
129 setwd("/Users/maia/Desktop/Thesis/individual/alt2")
130 lapply(1:length(GPS_bybuff), function(i) write.csv(GPS_
    bybuff[[i]], file = paste0(names(GPS_bybuff[i]), "alt2.
    csv"), row.names = FALSE))
131
132 setwd("/Users/maia/Desktop/Thesis/individual/alt3")
133 lapply(1:length(GPS_bybuff), function(i) write.csv(GPS_
    bybuff[[i]], file = paste0(names(GPS_bybuff[i]), "alt3.
    csv"), row.names = FALSE))
```

Appendix B

I used ArcGIS to transform the step files into spatial line objects. Since there were $36 \text{ buffalo} \times 4 \text{ step files} = 144 \text{ files}$, I used a model to process them automatically. The final step, XY To Line, transforms the table into lines using the start and end points.



Appendix C

I created data tables for each step file using the `rgdal` and `raster` packages.

```
1 ### Read in Buffalo Steps Data (Used and Alternative Steps)
   ###
2 ### Add Environmental Variable Data for each Step ###
3
4 # Projection used is UTM Zone 34S
5 crsref <- CRS("+proj=utm +zone=34 +south +datum=WGS84 +
   units=m +no_defs +ellps=WGS84 +towgs84=0,0,0")
6
7 # Read in shapefiles for fences, rivers, roads, waterholes
8 fences <- readOGR("/Users/maia/Desktop/Thesis/regisdata", "
   Fence")
9 rivers <- readOGR("/Users/maia/Desktop/Thesis/regisdata", "
   River")
10 roads <- readOGR("/Users/maia/Desktop/Thesis/regisdata", "
   Road")
11 waterholes <- readOGR("/Users/maia/Desktop/Thesis/regisdata
   ", "Water")
12
13
14 setwd("/Users/maia/Desktop/Thesis/Maia2/shp")
15
16 # Prepare list of buffalo step files by name
17 tracknames <- read.table("test.txt")
18 tracknames <- as.vector(tracknames$V1[seq(1, 1153-8, 8)])
19 tracknames <- strtrim(tracknames, nchar(tracknames)-4)
20
21 # Loop to read in each buffalo step file and add column
   with fence, river, road, waterhole data
22 tracks <- list()
23 for (t in tracknames){
24
```

```

25 track <- readOGR(dsn = ".", layer = t)
26
27 g <- gIntersects(track, fences, byid=T)
28 f <- colSums(g)
29 track@data$Fence <- f
30
31 g <- gIntersects(track, rivers, byid=T)
32 ri <- colSums(g)
33 track@data$River <- ri
34
35 g <- gIntersects(track, roads, byid=T)
36 ro <- colSums(g)
37 track@data$Road <- ro
38
39 g <- gIntersects(track, waterholes, byid=T)
40 w <- colSums(g)
41 track@data$Waterholes <- w
42
43 tracks <- c(tracks, track)
44
45 }
46
47
48
49 ### Add Treecover Data for each Step ###
50
51 # Read in treecover and project to UTM 34S
52 treecover <- new("GDALReadOnlyDataset", "/Users/maia/
    Desktop/Thesis/gisstuff/tree_frac_mosaic_GCS.tif")
53 treecover <- treecover[1:1270,1:3107]
54 treecover$band1[treecover$band1==200]<-0 #converting '200'
    water values to 0
55 treecover <-raster(treecover)

```

```

56 treecover <- projectRaster(treecover, crs=crsref)
57
58 # Parallelization for speed (even with this, it takes a few
    hours)
59 library(doParallel)
60 library(foreach)
61
62 no_cores <- detectCores() - 1
63 cl <- makeCluster(no_cores)
64 registerDoParallel(cl)
65
66 # Calculate treecover for each buffalo step file
67 treevalues <- foreach(i=1:116, .combine=cbind, .packages='
    raster') %dopar% {
68
69   trees <- extract(treecover, tracks[[i]], fun = mean)
70   trees
71
72 }
73
74 stopCluster(cl)
75
76 write.csv(treevalues, file = "/Users/maia/Desktop/Thesis/
    treevalues.csv") # Save results in separate CSV
77
78 # Add treecover columns to data
79 for (i in c(1:116)){
80   tracks[[i]]@data$Treecover <- treevalues[1:nrow(tracks[[i]
    ]@data), i]
81 }
82
83 # Do files 117-120 separately because they don't work in
    the for loop for some reason (they are the largest)

```

```

84 trees117 <- extract(treecover, tracks[[117]], fun = mean)
85 trees118 <- extract(treecover, tracks[[118]], fun = mean)
86 trees119 <- extract(treecover, tracks[[119]], fun = mean)
87 trees120 <- extract(treecover, tracks[[120]], fun = mean)
88
89 tracks[[117]]@data$Treecover <- trees117
90 tracks[[118]]@data$Treecover <- trees118
91 tracks[[119]]@data$Treecover <- trees119
92 tracks[[120]]@data$Treecover <- trees120
93
94 # Calculate treecover for the rest
95 no_cores <- detectCores() - 1
96 cl <- makeCluster(no_cores)
97 registerDoParallel(cl)
98
99 treevalues <- foreach(i=121:144, .combine=cbind, .packages=
    'raster') %dopar% {
100
101   trees <- extract(treecover, tracks[[i]], fun = mean)
102   trees
103
104 }
105
106 stopCluster(cl)
107
108 write.csv(treevalues, file = "/Users/maia/Desktop/Thesis/
    treevalues2.csv")
109
110 for (i in c(121:144)){
111   tracks[[i]]@data$Treecover <- treevalues[1:nrow(tracks[[i
    ]@data), i-120]
112 }
113

```

```

114
115
116 ### Add Woodland, Floodplain, Grassland, Human, Bushland
      Data for each Step ###
117
118 # Read in landcover raster
119 landscape <- raster("/Users/maia/Desktop/Thesis/Landcover1/
      Landcover1.tif")
120
121 # Reclassify woodland as 1, else as 0
122 m <- c(0, 6.5, 0, 6.5, 8.5, 1, 8.5, 14.5, 0, 14.5, 15.5,
      1, 15.5, 30, 0)
123 reclass <- matrix(m, ncol=3, byrow=TRUE)
124 woodland <- reclassify(landscape, reclass)
125
126 # Calculate woodland for each buffalo step file using
      parallelization
127 cl <- makeCluster(no_cores)
128 registerDoParallel(cl)
129 woodlandvalues <- foreach(i=1:116, .combine=cbind, .
      packages='raster') %dopar% {
130
131   w <- extract(woodland, tracks[[i]], fun = mean)
132   w
133
134 }
135 stopCluster(cl)
136
137 write.csv(woodlandvalues, file = "/Users/maia/Desktop/
      Thesis/woodlandvalues.csv")
138
139 for (i in c(1:116)){
140   tracks[[i]]@data$Woodland <- woodlandvalues[1:nrow(tracks

```

```

        [[i]]@data), i]
141 }
142
143 woodland <- crop(woodland, extent(treecover))
144 wood117 <- extract(woodland, tracks[[117]], fun = mean)
145 wood118 <- extract(woodland, tracks[[118]], fun = mean)
146 wood119 <- extract(woodland, tracks[[119]], fun = mean)
147 wood120 <- extract(woodland, tracks[[120]], fun = mean)
148
149 tracks[[117]]@data$Woodland <- wood117
150 tracks[[118]]@data$Woodland <- wood118
151 tracks[[119]]@data$Woodland <- wood119
152 tracks[[120]]@data$Woodland <- wood120
153
154 cl <- makeCluster(no_cores)
155 registerDoParallel(cl)
156 woodlandvalues2 <- foreach(i=121:144, .combine=cbind, .
      packages='raster') %dopar% {
157
158   w <- extract(woodland, tracks[[i]], fun = mean)
159   w
160
161 }
162 stopCluster(cl)
163
164 write.csv(woodlandvalues2, file = "/Users/maia/Desktop/
      Thesis/woodlandvalues2.csv")
165
166 for (i in c(121:144)){
167   tracks[[i]]@data$Woodland <- woodlandvalues2[1:nrow(
      tracks[[i]]@data), i-120]
168 }
169

```



```

170
171 # Repeat for bushland
172 m <- c(0, 16.5, 0, 16.5, 17.5, 1, 17.5, 30, 0)
173 reclass <- matrix(m, ncol=3, byrow=TRUE)
174 bushland <- reclassify(landscape, reclass)
175
176 bushland <- crop(bushland, extent(treecover))
177
178 cl <- makeCluster(no_cores)
179 registerDoParallel(cl)
180 bushlandvalues <- foreach(i=1:116, .combine=cbind, .
      packages='raster') %dopar% {
181
182   b <- extract(bushland, tracks[[i]], fun = mean)
183   b
184
185 }
186 stopCluster(cl)
187
188 write.csv(bushlandvalues, file = "/Users/maia/Desktop/
      Thesis/bushlandvalues.csv")
189
190 for (i in c(1:116)){
191   tracks[[i]]@data$Bushland <- bushlandvalues[1:nrow(tracks
      [[i]]@data), i]
192 }
193
194 bush117 <- extract(bushland, tracks[[117]], fun = mean)
195 bush118 <- extract(bushland, tracks[[118]], fun = mean)
196 bush119 <- extract(bushland, tracks[[119]], fun = mean)
197 bush120 <- extract(bushland, tracks[[120]], fun = mean)
198
199 tracks[[117]]@data$Bushland <- bush117

```

```

200 tracks[[118]]@data$Bushland <- bush118
201 tracks[[119]]@data$Bushland <- bush119
202 tracks[[120]]@data$Bushland <- bush120
203
204 cl <- makeCluster(no_cores)
205 registerDoParallel(cl)
206 bushlandvalues2 <- foreach(i=121:144, .combine=cbind, .
      packages='raster') %dopar% {
207
208   b <- extract(bushland, tracks[[i]], fun = mean)
209   b
210
211 }
212 stopCluster(cl)
213
214 write.csv(bushlandvalues2, file = "/Users/maia/Desktop/
      Thesis/bushlandvalues2.csv")
215
216 for (i in c(121:144)){
217   tracks[[i]]@data$Bushland <- bushlandvalues2[1:nrow(
      tracks[[i]]@data), i-120]
218 }
219
220
221 # Repeat for grassland
222 m <- c(0, 17.5, 0, 17.5, 18.5, 1, 18.5, 30, 0)
223 reclass <- matrix(m, ncol=3, byrow=TRUE)
224 grassland <- reclassify(landscape, reclass)
225
226 grassland <- crop(grassland, extent(treecover))
227
228 cl <- makeCluster(no_cores)
229 registerDoParallel(cl)

```

```

230 grasslandvalues <- foreach(i=1:116, .combine=cbind, .
      packages='raster') %dopar% {
231
232   g <- extract(grassland, tracks[[i]], fun = mean)
233   g
234
235 }
236 stopCluster(cl)
237
238 write.csv(grasslandvalues, file = "/Users/maia/Desktop/
      Thesis/grasslandvalues.csv")
239
240 for (i in c(1:116)){
241   tracks[[i]]@data$Grassland <- grasslandvalues[1:nrow(
      tracks[[i]]@data), i]
242 }
243
244 grass117 <- extract(grassland, tracks[[117]], fun = mean)
245 grass118 <- extract(grassland, tracks[[118]], fun = mean)
246 grass119 <- extract(grassland, tracks[[119]], fun = mean)
247 grass120 <- extract(grassland, tracks[[120]], fun = mean)
248
249 tracks[[117]]@data$Grassland <- grass117
250 tracks[[118]]@data$Grassland <- grass118
251 tracks[[119]]@data$Grassland <- grass119
252 tracks[[120]]@data$Grassland <- grass120
253
254 cl <- makeCluster(no_cores)
255 registerDoParallel(cl)
256 grasslandvalues2 <- foreach(i=121:144, .combine=cbind, .
      packages='raster') %dopar% {
257
258   g <- extract(grassland, tracks[[i]], fun = mean)

```

```

259   g
260
261 }
262 stopCluster(cl)
263
264 write.csv(grasslandvalues2, file = "/Users/maia/Desktop/
      Thesis/grasslandvalues2.csv")
265
266 for (i in c(121:144)){
267   tracks[[i]]@data$Grassland <- grasslandvalues2[1:nrow(
      tracks[[i]]@data), i-120]
268 }
269
270
271 # Repeat for anthropogenic land use
272 m <- c(0, 8.5, 0, 8.5, 12.5, 1, 12.5, 30, 0)
273 reclass <- matrix(m, ncol=3, byrow=TRUE)
274 human <- reclassify(landscape, reclass)
275
276 human <- crop(human, extent(treecover))
277
278 cl <- makeCluster(no_cores)
279 registerDoParallel(cl)
280 humanvalues <- foreach(i=1:116, .combine=cbind, .packages='
      raster') %dopar% {
281
282   h <- extract(human, tracks[[i]], fun = mean)
283   h
284
285 }
286 stopCluster(cl)
287
288 write.csv(humanvalues, file = "/Users/maia/Desktop/Thesis/

```

```

    humanvalues.csv")
289
290 for (i in c(1:116)){
291   tracks[[i]]@data$Human <- humanvalues[1:nrow(tracks[[i]]
      @data), i]
292 }
293
294 human117 <- extract(human, tracks[[117]], fun = mean)
295 human118 <- extract(human, tracks[[118]], fun = mean)
296 human119 <- extract(human, tracks[[119]], fun = mean)
297 human120 <- extract(human, tracks[[120]], fun = mean)
298
299 tracks[[117]]@data$Human <- human117
300 tracks[[118]]@data$Human <- human118
301 tracks[[119]]@data$Human <- human119
302 tracks[[120]]@data$Human <- human120
303
304 cl <- makeCluster(no_cores)
305 registerDoParallel(cl)
306 humanvalues2 <- foreach(i=121:144, .combine=cbind, .
      packages='raster') %dopar% {
307
308   h <- extract(human, tracks[[i]], fun = mean)
309   h
310
311 }
312 stopCluster(cl)
313
314 write.csv(humanvalues2, file = "/Users/maia/Desktop/Thesis/
      humanvalues2.csv")
315
316 for (i in c(121:144)){
317   tracks[[i]]@data$Human <- humanvalues2[1:nrow(tracks[[i]]

```

```

        @data), i-120]
318 }
319
320
321 # Repeat for floodplain
322 m <- c(0, 4.5, 0, 4.5, 5.5, 1, 5.5, 30, 0)
323 reclass <- matrix(m, ncol=3, byrow=TRUE)
324 floodplain <- reclassify(landscape, reclass)
325
326 floodplain <- crop(floodplain, extent(treecover))
327
328 cl <- makeCluster(no_cores)
329 registerDoParallel(cl)
330 floodplainvalues <- foreach(i=1:116, .combine=cbind, .
        packages='raster') %dopar% {
331
332   f <- extract(floodplain, tracks[[i]], fun = mean)
333   f
334
335 }
336 stopCluster(cl)
337
338 write.csv(floodplainvalues, file = "/Users/maia/Desktop/
        Thesis/floodplainvalues.csv")
339
340 for (i in c(1:116)){
341   tracks[[i]]@data$Floodplain <- floodplainvalues[1:nrow(
        tracks[[i]]@data), i]
342 }
343
344 flood117 <- extract(floodplain, tracks[[117]], fun = mean)
345 flood118 <- extract(floodplain, tracks[[118]], fun = mean)
346 flood119 <- extract(floodplain, tracks[[119]], fun = mean)

```

```

347 flood120 <- extract(floodplain, tracks[[120]], fun = mean)
348
349 tracks[[117]]@data$Floodplain <- flood117
350 tracks[[118]]@data$Floodplain <- flood118
351 tracks[[119]]@data$Floodplain <- flood119
352 tracks[[120]]@data$Floodplain <- flood120
353
354 cl <- makeCluster(no_cores)
355 registerDoParallel(cl)
356 floodplainvalues2 <- foreach(i=121:144, .combine=cbind, .
      packages='raster') %dopar% {
357
358   f <- extract(floodplain, tracks[[i]], fun = mean)
359   f
360
361 }
362 stopCluster(cl)
363
364 write.csv(floodplainvalues2, file = "/Users/maia/Desktop/
      Thesis/floodplainvalues2.csv")
365
366 for (i in c(121:144)){
367   tracks[[i]]@data$Floodplain <- floodplainvalues2[1:nrow(
      tracks[[i]]@data), i-120]
368 }

```

Appendix D

I calculated the beta coefficients using the data from each buffalo's used and alternative steps, using the `mcllogit` package in R. Then, I used a t-test for each variable's betas to determine which were different from 0 at the 5% significance level. Finally, I made a population-level model using only the significant variables. I evaluated the model by comparing the log-likelihood using the model to the null model for each individual buffalo.

```
1 betas <- data.frame(fe=double(), ri=double(), ro=double(),
2   wa=double(), tr=double(), wo=double(), bu=double(), gr=
3   double(), hu=double(), fl=double(), stringsAsFactors=
4   FALSE)
5
6 # Loop through each buffalo, create a conditional logistic
7   model, and save coefficients into a matrix for analysis
8
9 for (i in seq(1,141,by=4)){
10   u <- tracks[[i]]@data[,6:15]
11   u$chosen <- rep(1,nrow(u))
12   u$index <- 1:nrow(u)
13
14   a1 <- tracks[[i+1]]@data[,6:15]
15   a1$chosen <- rep(0,nrow(a1))
16   a1$index <- 1:nrow(a1)
17
18   a2 <- tracks[[i+2]]@data[,6:15]
19   a2$chosen <- rep(0,nrow(a2))
20   a2$index <- 1:nrow(a2)
21
22   a3 <- tracks[[i+3]]@data[,6:15]
23   a3$chosen <- rep(0,nrow(a3))
24   a3$index <- 1:nrow(a3)
25
26   test <- rbind(u, a1, a2, a3)
```



```

22
23   coefs <- mclogit(cbind(test$chosen,test$index) ~ Fence +
      River + Road + Waterholes + Treecover + Woodland +
      Bushland + Grassland + Human + Floodplain, data =test)
      $coefficients
24
25   coefs <- as.data.frame(coefs)
26
27   betas <- rbind.fill(betas, as.data.frame(t(coefs)))
28 }
29
30 # t-test for each variable
31 t.test(betas$Fence)
32 t.test(betas$River)
33 t.test(betas$Road)
34 t.test(betas$Waterholes)
35 t.test(betas$Treecover)
36 t.test(betas$Woodland)
37 t.test(betas$Bushland)
38 t.test(betas$Grassland)
39 t.test(betas$Human)
40 t.test(betas$Floodplain)
41
42 # Combine all buffalo steps into one used dataset and three
      alternative datasets
43 used <- tracks[[1]]@data[,6:15]
44 for (i in c
      (5,9,13,14,21,25,29,33,37,41,45,53,57,61,65,69,73,77,
      81,85,89,93,101,121,125,129,133,137,141)){
45   u <- tracks[[i]]@data[,6:15]
46   used <- rbind(used,u)
47 }
48

```

```

49 alt1 <- as.matrix(tracks[[2]]@data[,6:15])
50 for (i in c
      (6,10,15,18,22,26,30,34,38,42,46,54,58,62,66,70,74,78,
      82,86,90,94,102,122,126,130,134,138,142)){
51   a1 <- tracks[[i]]@data[,6:15]
52   alt1 <- rbind(alt1,a1)
53 }
54
55 alt2 <- as.matrix(tracks[[3]]@data[,6:15])
56 for (i in c
      (7,11,16,19,23,27,31,35,39,43,47,55,59,63,67,71,75,79,
      83,87,91,95,103,123,127,131,135,139,143)){
57   a2 <- tracks[[i]]@data[,6:15]
58   alt2 <- rbind(alt2,a2)
59 }
60
61 alt3 <- as.matrix(tracks[[4]]@data[,6:15])
62 for (i in c
      (8,12,17,20,24,28,32,36,40,44,48,56,60,64,68,72,76,80,
      84,88,92,96,104,124,128,132,136,140,144)){
63   a3 <- tracks[[i]]@data[,6:15]
64   alt3 <- rbind(alt3,a3)
65 }
66
67 # Steps are matched by index. If they are used, they get a
   "1" in the chosen column, else "0"
68 used$chosen <- rep(1,nrow(used))
69 used$index <- 1:nrow(used)
70
71 alt1$chosen <- rep(0,nrow(alt1))
72 alt1$index <- 1:nrow(alt1)
73
74 alt2$chosen <- rep(0,nrow(alt2))

```

```

75 alt2$index <- 1:nrow(alt2)
76
77 alt3$chosen <- rep(0,nrow(alt3))
78 alt3$index <- 1:nrow(alt3)
79
80 test <- rbind(used, alt1, alt2, alt3)
81
82 # Population-level model using Fence, River, and Human as
      predictors because they were significant in the t-test
      across individuals
83 total <- mclogit(cbind(test$chosen, test$index) ~ Fence +
      River + Human, data = test)
84
85 # Calculation of log-likelihood difference
86
87 neglogliks <- vector()
88
89 for (i in seq(1,141,by=4)){
90   nLL <- function(fe, ri, hu) {
91     B <- as.vector(c(fe, ri, hu))
92     used <- as.matrix(tracks[[i]]@data[,c(6,7,14)])
93     alt1 <- as.matrix(tracks[[i+1]]@data[,c(6,7,14)])
94     alt2 <- as.matrix(tracks[[i+2]]@data[,c(6,7,14)])
95     alt3 <- as.matrix(tracks[[i+3]]@data[,c(6,7,14)])
96
97     func <- function(x){
98       return (-log(exp(B %*% used[x,]) + exp(B %*% alt1[x
99         ,]) + exp(B %*% alt2[x,]) + exp(B %*% alt3[x,])))
100     }
101
102     LL <- sum(apply(used, 1, function(x) B %*% x)) + sum(
      sapply(seq(length=nrow(used)), func))
102   return (-LL)

```

```

103   }
104
105   neglogliks <- c(neglogliks, nLL(0,0,0) - nLL(-1.27,
      -3.06, -2.1))
106 }
107
108 # Plot the differences
109 neglogliks <- as.data.frame(neglogliks)
110 g <- ggplot(neglogliks, aes(y = neglogliks, x = seq(1,
      length(neglogliks)))) + geom_point() +
111     labs(x="Individual Buffalo", y="Improvement
      in Log-likelihood") +
112 theme(axis.text.x = element_blank()) + theme(plot.title =
      element_text(hjust = 0.5))

```