**Lab 1: GPS Data Cleaning and Visualization**

**Managing GPS data**

Radio tracking wildlife is increasingly being used to understand spatial needs and/or behavioral pattern of animals. Increasingly, collars are fitted with GPS units allowing the recording of precise information on the location of an animal at a given time. In using GPS tracking to collect information on animals, individuals first step is to decide on how they want to sample animal locations. This decision is typically a trade-off between data resolution (number of fixes per hour/day) versus tracking period (number of weeks/months/years for which data are collected) which are bounded by the battery capacity (i.e. number of fixes) and, at times, life of the collar unit itself. We have to choose to collect fine resolution data on movement (short fix intervals) for a short duration or coarser data on animal locations (but typically not movement per say) for longer intervals. The fix schedule one chooses is strongly dependent on the questions one would like to ask of the data.

Once the GPS unit has been deployed on an animal and the data collected, there are several important steps that must be taken prior to analyzing the data. First, we must ensure the collar data is in local time. Often the collar is preprogramed to record at the local time where the collar company is located or at 0 UCT (Coordinated Universal Time). Our first step will be to load the data in R, and ensure the time is the correct local time where the data were collected. Then,

Secondly, it is critical the data are scrutinized for errors and those errors are removed prior to analyzing the tracking data. In this lab, we will focus on protocols to clean collected GPS data. This includes: (1) Removal of fixes that happen outside the period that the collar was placed on the animal (typically before the collar is deployed or after it falls off the animal); (2) Removal of incomplete records (i.e. those that date, time, latitude or longitude are missing; (3) removal of all repeated/double recordings; (4) Removal of fixes with unacceptably high error rates as registered from the dilution of precision (DOP) information; (5) Removal of positions that represent biologically implausible speeds of movements (e.g. fixes that show the animal moved incredibly fast as can happen if the position of GPS location is some distance from where an animal actually is).

Once the data is clean, we will learn how to align and extract spatial covariate data with GPS telemetry data using Google Earth Engine. We will use this spatial data to look at patterns of resource use by time of day. Finally, we will create animations of the movement data. Throughout the lab, we will show you different ways to visualize the data spatially and temporally to aid in the cleaning and preliminary analysis process.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

The code used for the GPS data cleaning is relatively advanced, so it has all been provided for you. There is code to remove fixes, visualize the data and summarize the data to allow biological insights. You can use this code by running through the script line-by-line. Be sure to read the top section of the code file before proceeding.

Copy the files in the Lab 1 folder into a folder of your choosing. You will have to assign this folder as your wording directory in the R code provided, so be clear where you have assigned these data. To make this process easier, we recommend creating an R Project in R Studio for the course that will help set the working directory for you. See the RStudio site for more information: [Using RStudio Projects – RStudio Support](https://support.rstudio.com/hc/en-us/articles/200526207-Using-Projects)

We are going to use a program called R Studio in this lab, as it allows you to see the different parameters you have assigned. We will begin by opening R Studio. To start this lab, we want to open the R script file in your Lab 1 folder called “Lab1\_GPSdataManagement.R”. You will see the code for this lab open in the primary panel in the upper left hand corner. The left lower panel is the actual R consul, where the code will be implemented. The lower right panel contains the libraries (i.e. programs) that are being used to run the code, as well as the panel to visualize your figures. Finally, the top right panel will contain the different variables you assign using the code. This can help you understand the variables you have assigned.

To get comfortable with R, we will play with a couple basic features before stepping into the code. In the lower left R console panel, type in a line of code (for example 5 + 5), and hit 'enter' to have R tell you the answer. This is basically how we will use the program. R works in an object-based framework. This means you can save and store information in different object types. To illustrate this type in the following code:

x = 5 + 5

x

If you did this correctly, then when you type in 'x' and hit 'enter' R should return a value of 10. The first line of code told R that you wanted x to be an object that was equal to 5 + 5. You can now use x in other calculations, for example, x \* 2 or x^2. If you look in the upper right panel, you will see x as the first object you have defined. We will use these objects to store different types of data, like the GPS locations of the elephants. The commands you type are actually a simple program. In this lab, we will use more complicated programs that have been coded by others and which we can access through the use of "libraries". Because R is open source anyone can create a program that can be stored and shared online in libraries. We can load them onto our computer to make use of them.

To initiate the code in the Lab1\_GPSdataManagement R file that should now be loaded in the top left panel, you can either (1) select the code and press “ctrl” and “enter” simultaneously or (2) click the Run button.

As we progress through the lab, questions are imbedded in the code to help you think through the data cleaning process.