Generation of Simulated Data and Model Accuracy

Josh Cullen

10 January 2020

Background

As part of the upcoming manuscript that compares our newly developed method against frequently used methods, we will be testing them using both simulated and empirical data. These tests on simulated tracks will be performed to identify and recover pre-specified activity centers (ACs) and pre-specified behavioral states. There are a wide variety of options available in terms of simulating animal movement, whether based on empirical distributions or by using parametric distributions. In other cases, simulations can be generated from a process model that already has built in constraints (hidden Markov models via the *momentuHMM* package).

To make the method comparison as unbiased as possible, I decided to use methods to simulate data that were only based on a correlated random walk (CRW). I initially considered using one of three different R packages to perform these simulations: *adehabitatLT, trajr, waddle*.

The *adehabitatLT* package is widely used for evaluating trajectories and also for simulating paths based on empirical distributions of step lengths (SL) and turning angles (TA). However, this package did not have a convenient method for producing multistate trajectories, which would be used to simulate changes in behavior, or for simulating based on a biased CRW (BCRW) that could be used to incorporate attraction to multiple ACs. Additionally, the parametric distributions (SL: chi; TA: wrapped normal) from which these simulated data would be drawn are not used by any other common method for behavior classification. Therefore, this package was not used.

The *trajr* package also does not have the capability to generate a multistate trajectory or to include ACs as part of a BRW. Additionally, there is limited control over defining the distribution from which step lengths and turning angles are drawn. Therefore, the *trajr* package was not used.

The *waddle* package was used to generate simulated trajectories used to compare first passage time (FPT), behavioral change point analysis (BCPA), Bayesian partitioning of Markov models (BPMM), and multistate random walks (MRW) in the publication by Gurarie et al. (2016). This package provides option for generating simulated data from a BCRW or correlated velocity movement (CVM) model, as well as multiphase extensions of these models. However, there was still limited control over movement parameters and data were returned as complex as opposed to real numbers. Therefore, this package also was not used to simulate animal movement.

Instead, I developed my own code to generate a multiphase CRW model of three different behaviors as well as a multiphase BRW that could be more easily controlled and produced locations as real numbers. These models were based upon established CRW and BRW models, but were modified to achieve the desired result.

Simulations

BRW for identifying ACs

First, values will be chosen for the number of observations per time segment (n), the number and location of ACs (Z.centers), the number of phases or time segments (nphases), the initial location (Z0), the shape (a) and scale (b) parameters for a gamma distribution from which to draw step lengths, and the concentration parameter (ρ) from which to draw the turning angles:

CRW for identifying behaviors

Since our model uses a mixed-membership model to identify the proportion of behaviors within each time segment, I simulated tracks using both hard-clustering and mixed-membership methods to test the model's ability to classify these different types of data. Both simulations required the user to define the length of each time segment (n), provide a vector of the behaviors (behav), a vector of the shape and scale parameters used to generate step lengths (SL.params), a vector of the mean angles and concentration parameters used to generate the turning angles (TA.params), and the initial location (Z0). A gamma distribution was used to generate step lengths while a wrapped Cauchy distribution was used to generate turning angles. For the hard-clustering method, the 'behav' vector is of length equal to the number of time segments where there is a greater probability of being in either a 'resting' or 'exploratory' behavior than in a 'transit' behavior. In the mixed membership model, this first step is the same, but another one is performed within each time segment where the original behavior assigned to each segment has the greatest probability of occurrence compared to the other two.

Hard-clustering

First, I will simulate data using the hard-clustering method for behavior and subsequently run it on the model to evaluate accuracy of behavior classification.

```
#define behaviors and randomly sample 50 (for 50 time segments)
#weight probs so that behavior 1 (Resting) occurs 50%, behavior 2 (Exploratory) occurs 35%, and behavio
set.seed(1)
behav<- sample(c(1,2,3), 50, replace = TRUE, prob = c(0.5, 0.35, 0.15))
n=50
SL.params<- data.frame(shape=c(0.25, 2, 10), scale = c(1, 1, 1))
TA.params<- data.frame(mu=c(pi, pi, 0), rho = c(0.8, 0.4, 0.8))

track<- CRW.sim(n=n, behav = behav, SL.params = SL.params, TA.params = TA.params, Z0=c(0,0))
```

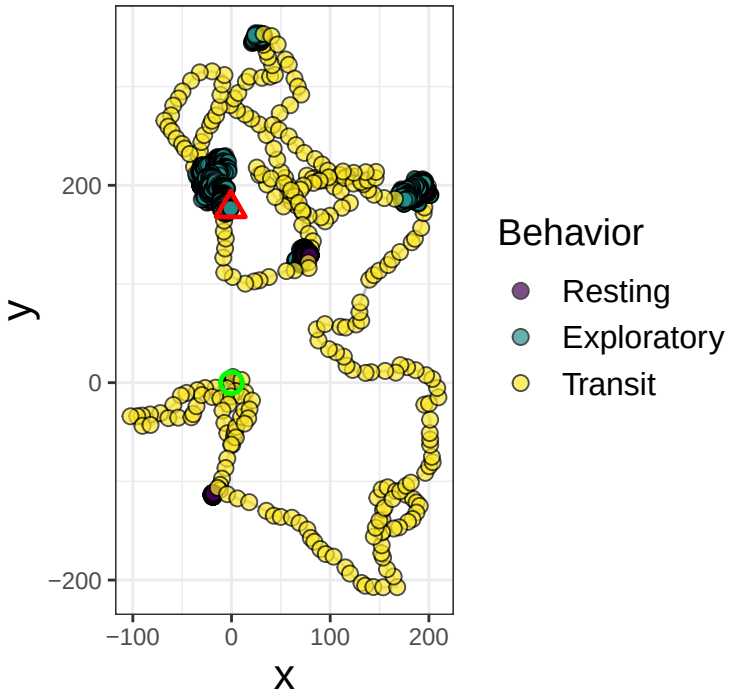

Figure 1: Simulated CRW track with a single behavior assigned for each time segment.

We can also compare the SL and TA distributions among behaviors:

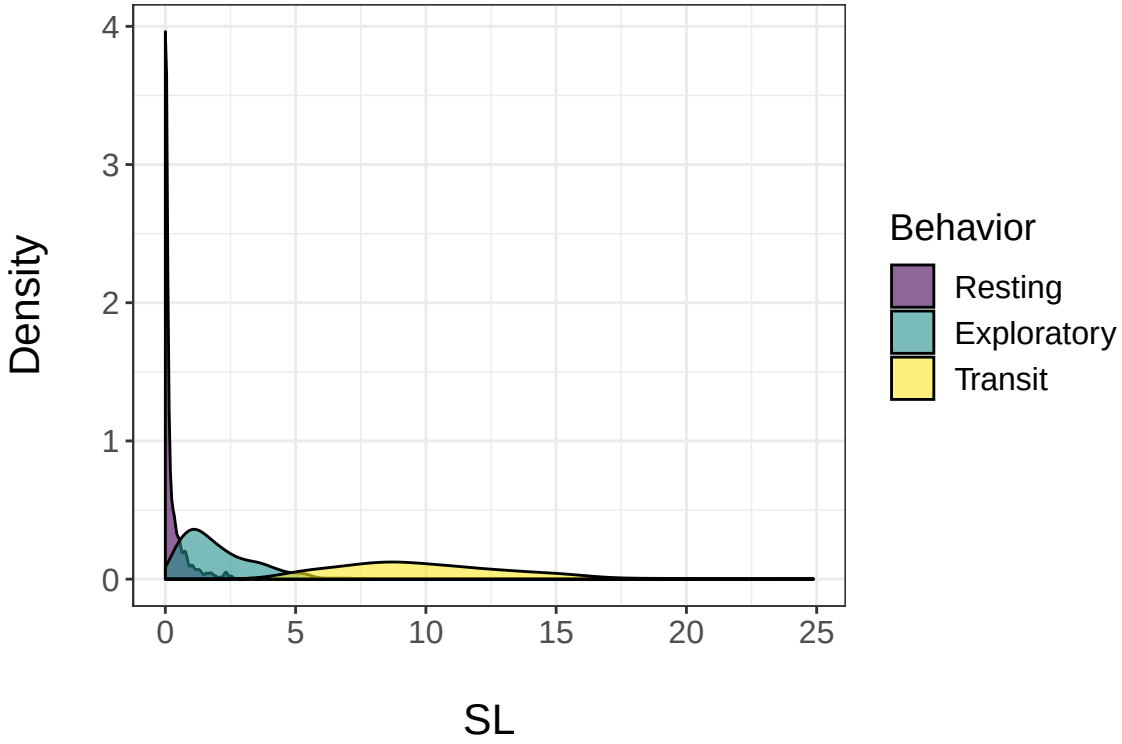

Figure 2: Distributions of step lengths.

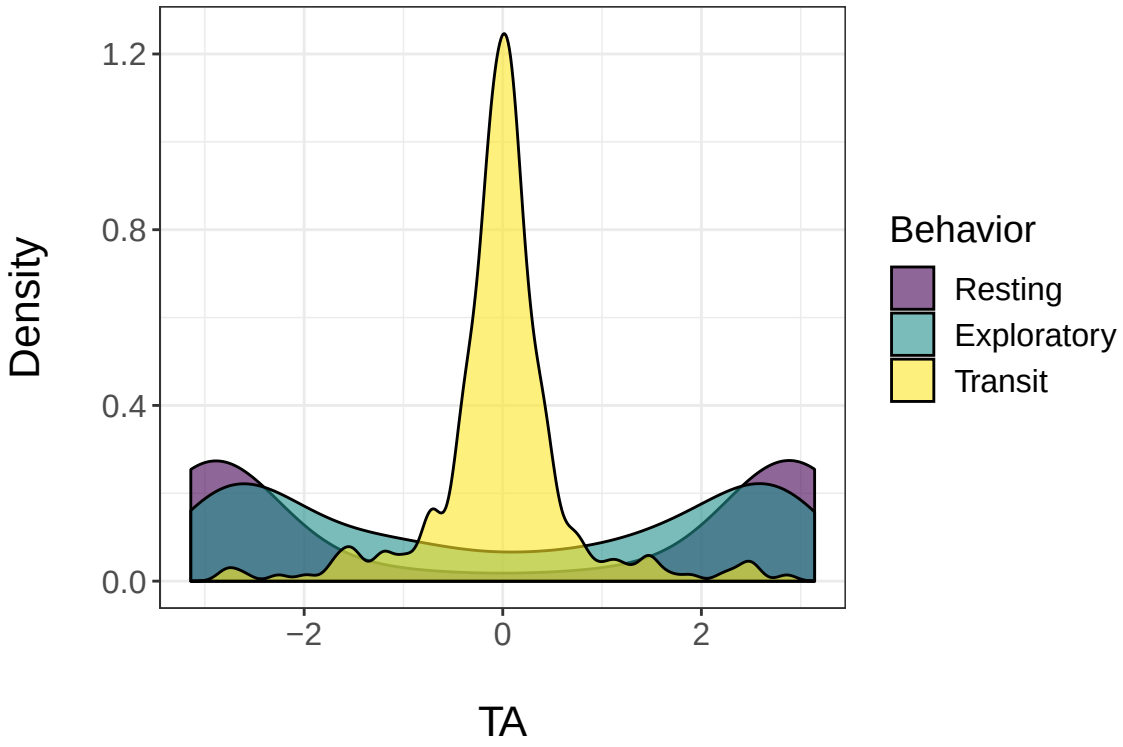

Figure 3: Distributions of turning angles.

From this simulated track, we can run the time segmentation algorithm to discern the number and locations of the breakpoints in the time series of SL and TA. These breakpoints are lined up with the data, as well as compared against the true breakpoints: