Generation of Simulated Data

Josh Cullen

17 December 2019

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 will be performed as 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 to generate real data. In other cases, simulations can be generated from a process model that already has built in constraints on what is being simulated (hidden Markov models via the *moveHMM* 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) or some version of this. 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 BCRW. 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 study generated three different simulations on which to compare the aforementioned methods, for which BCRW appeared to be the best option. This appeared to be the best option since it would allow the specification of the locations for ACs, as well as the attraction strength to these locations. Additionally, these BCRWs could be modeled in a multistate fashion, which would potentially permit the generation of a single simulated trajectory on which to test the model for identifying ACs in addition to behavioral states. However, there was limited capacity for controlling the Weibull distribution from which SLs are drawn and therefore this model could only be used to test the algorithms to identify ACs. The authors also used another model, which was a correlated velocity movement (CVM) model in continuous time based on a Ornstein-Uhlenbeck (OU) process. This (OU) process is essentially the continuous-time version of a CRW used in discrete-time frameworks. By using a multistate version of the CVM model, I could have greater control over changes in speed (SL) over each time segment, as well as the concentration parameter governing the distribution of TA. Therefore, two sets of simulated data were generated from the *waddle* package that could fulfill the needs for identifying ACs as well as behavioral states.
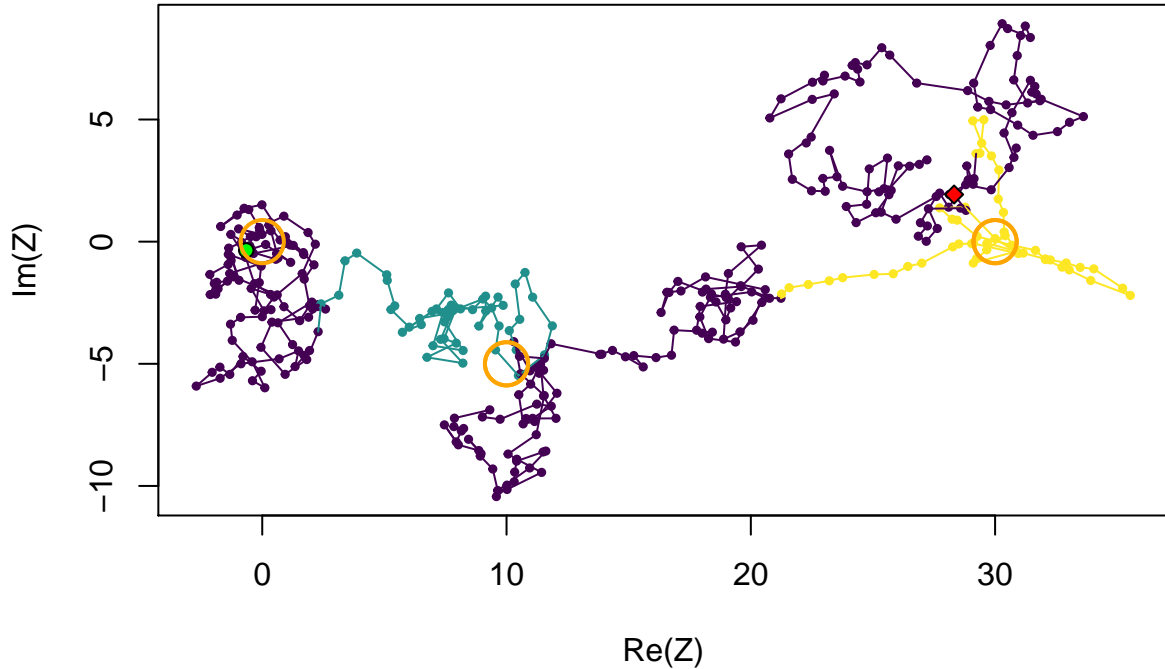
Figure 1: BCRW model. The different colors each represent time segments with different values for the concentration parameter on the turning angles, where purple is the most uniform and yellow is the most concentrated near an angle of 0 radians. Orange circles denote the locations of the three specified ACs.

Simulations

BCRW for identifying ACs

First, values will be chosen for the number of observations per time segment, the ACs, the strength of attraction towards ACs, and the concentration parameters of the TA distribution:

```r
library(waddle)
library(tidyverse)
library(viridis)

set.seed(1)

#multistate BCRW w 5 phases
ns <- c(100,50,100,50,100)   #N
Z.centers <- c(0,10-5i,10-5i,30,30)   #locations of ACs
attractions <- c(0.9,0.9,0.9,0.9,0.9)   #strength of attraction towards ACs
rhos <- c(0.2,0.4,0.2,0.9,0.2)   #concentration parameter for TA

BCRW.sim <- multiBCRW(rhos=rhos, attractions=attractions, Z.centers=Z.centers, ns=ns)
```

All time intervals are regular, so the SL is also representative of the speed for each relocation. Since the way this model is set up does not easily allow for changing the SL, all SL/speed should be roughly equal for all time segments:

Although there are not any differences in the SL across time segments, the concentration parameter ρ allows the user to determine whether TA is concentrated around 0 rad or is dispersed across all angles:
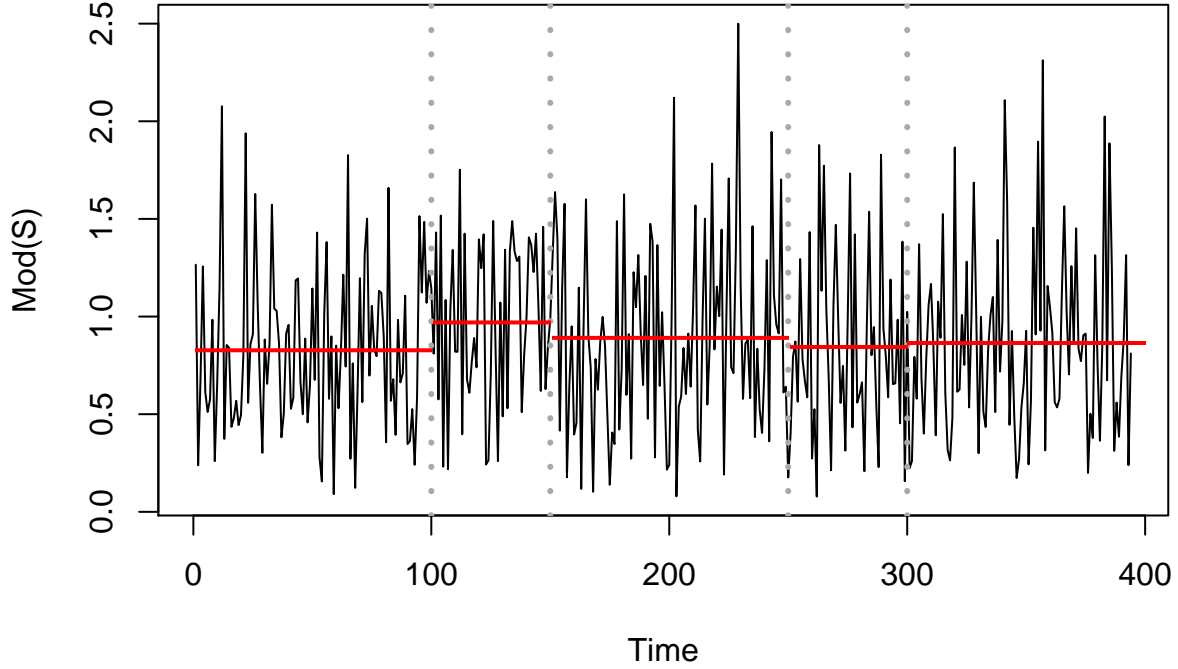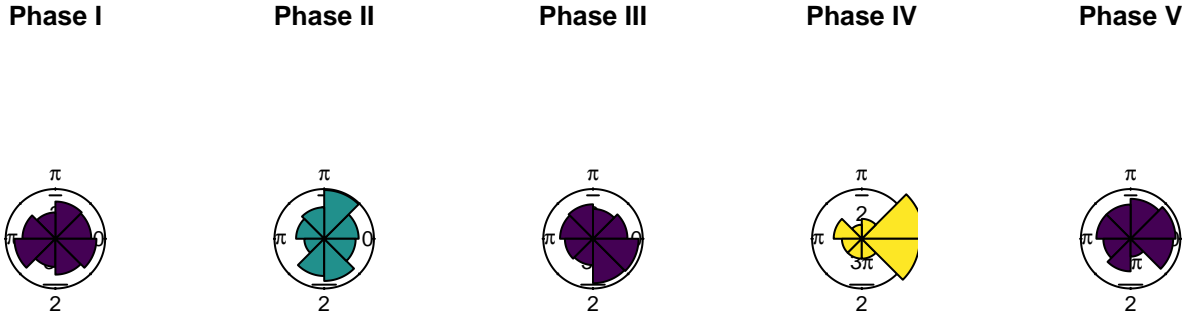
Figure 2: Dashed grey lines indicate the change between time segments and the red lines denote the mean value of the SL/speed by time segment.

Phase I   **Phase II**   **Phase III**   **Phase IV**   **Phase V**

CVM for identifying behaviors

Parameters governing the mean speed per time segment (ν) and the time-scale of autocorrelation (τ; or tortuosity) were defined for each time segment, for which the durations were defined by *Ts*. With this modeling framework, three behaviors were created by adjusting the mean speed and tortuosity (proxy for TA):

```
set.seed(10)
nus <- c(1,2.5,1,10,1)
taus <- c(1,1,1,10,1)
Ts <- c(100,50,100,50,100)
CVM.sim <- multiCVM(taus, nus, Ts)
plot(CVM.sim, col = viridis(n=3)[c(1,2,1,3,1)])
```

By having the power to control the mean speed, the step lengths did vary across these different behaviors:
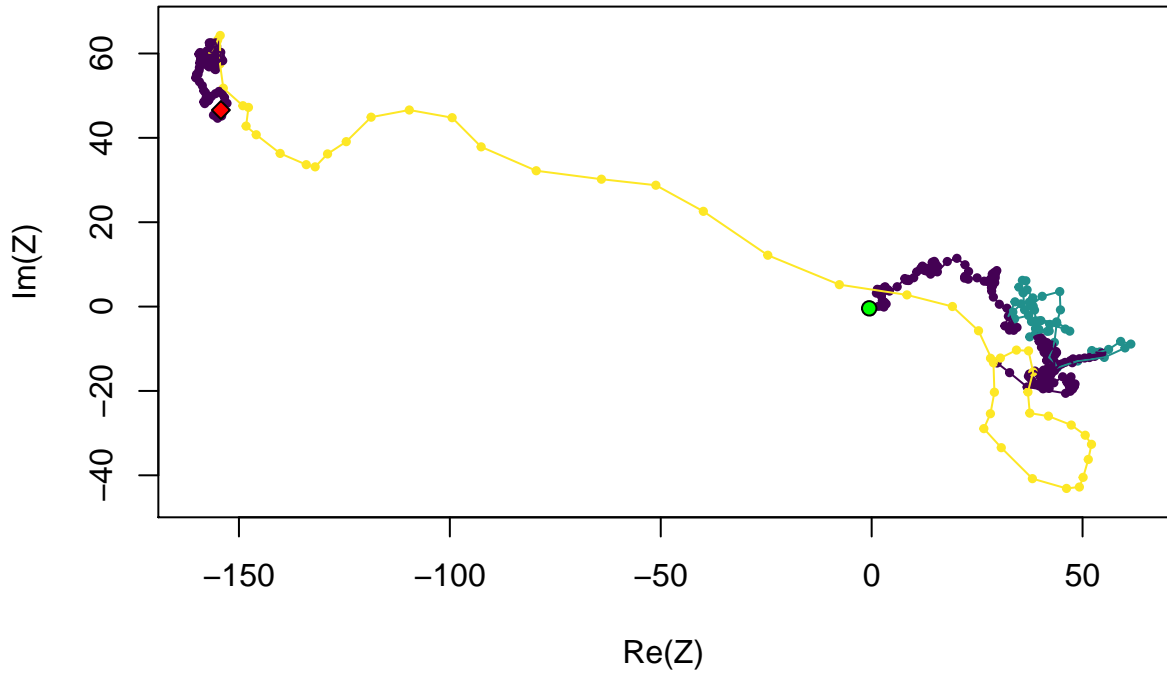
3

Figure 3: Colors denote different behavioral states based on mean speed and time-scale of autocorrelation in the correlated velocity movement model. Cooler to warmer colors represent a change from slower to faster behaviors for each of the five time segments.
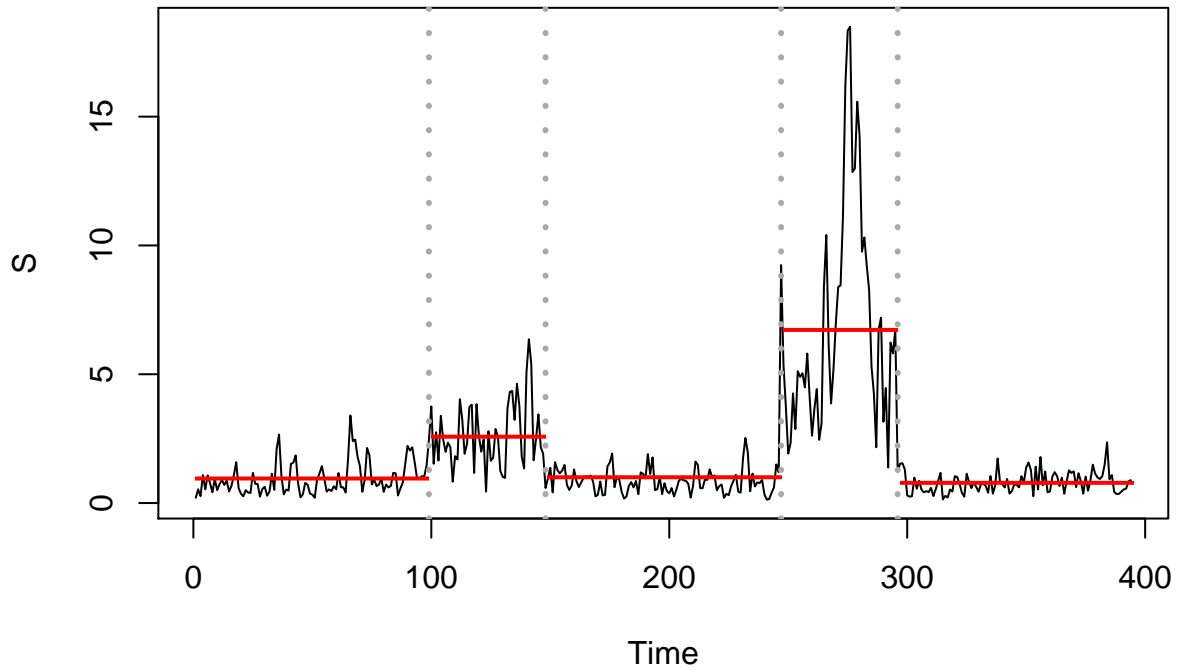

Figure 4: A trace of the speed per observation is shown for the regularized trajectory. Dashed grey lines indicate changes by time segment and red lines represent the mean speed per segment.

Similar to the concentration parameter ρ used in the BCRW model, τ controlled the turning angle for the CVM model:
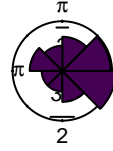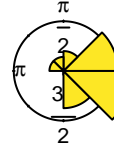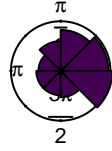
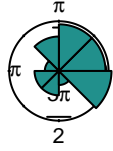| Phase I	Phase II	Phase III	Phase IV	Phase V
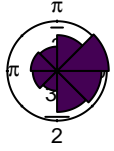
These models can be modified depending on the complexity we would like to include (e.g. longer tracks, more ACs, more behaviors, more trajectories, etc). Apart from coding our own BCRW model or using the restrictive framework of mechanistic movement models (i.e. HMMs) to generate simulated trajectories, this package appears to provide the best options available that I could find from R packages for animal movement.