Hidden Markov models in R

5 November 2019

These exercises offer an introduction to animal movement modelling using the R package moveHMM. They will take you through the different steps of a typical analysis, based on one track of elephant movement.

The package moveHMM can be installed and loaded in R with the following commands:

```
install.packages("moveHMM")
library(moveHMM)
```

Information on the package, and how to use it, is available in its vignette, and more technical details about each function are available in the documentation of the package (e.g., with `?fitHMM`).

1. *Load and prepare the data.*
   (a) Load the elephant movement data set ("elephant.csv") into a data frame.
   (b) Use the function `prepData` on the loaded data, to compute the step lengths and turning angles.
   (c) Visualise the data by calling the function `plot` on the data frame output by `prepData`.
2. *Fit an HMM to the data.*
   (a) Using the function `fitHMM`, fit a 2-state HMM to the elephant data, and store the returned model, e.g.

   ```
   myhmm <- fitHMM(...)
   ```

   Only four arguments are needed:

   - `data`, the data frame returned by `prepData`;
   - `nbStates`, the number of states (here, 2);
   - `stepPar0`, the initial parameter values for the step length distribution;
   - `anglePar0`, the initial parameter values for the turning angle distribution.

Histograms of the observed step lengths and turning angles can help with the choice of initial parameters, e.g.

```
hist(data$step)
hist(data$angle)
```

(b) Visualise the fitted model with `plot`. For more graphical options, see `?plot.moveHMM`.

(c) Visualise the pseudo-residuals of the model with `plotPR`. Based on the quantile-quantile plots of the pseudo-residuals against the normal distribution, does the model seem to fit the data well?

3. *Include covariates in the transition probabilities.*

With `fitHMM`, the argument `formula` can be used to include covariates in the transition probabilities. It should be a model formula, as expected by standard functions such as `lm`. For example,

```
formula = ~ cov1 # include covariate cov1
formula = ~ cov1 + cov2 # include covariates cov1 and cov2
formula = ~ cov1 * cov2 # include covariates cov1 and cov2 with interaction
```

(a) Fit an HMM to the elephant data, with the temperature ('`temp`') as a covariate.

(b) Fit an HMM to the elephant data, with the time of day ('`tod`') as a covariate.

(c) Visualise the fitted models, e.g. using `plot` and `plotStationary`. In both functions, you can use the argument `plotCI` to plot confidence intervals and visualise the uncertainty in the estimates. Do the covariates seem to have an effect on the transition probabilities?

(d) The AIC is a model selection criterion, to choose between competing model formulations. Here, we can use it to decide which covariate(s) to keep in the model. Use the function `AIC` to compare the three fitted models: (1) covariate-free model, (2) model with temperature, (3) model with time of day. Which model formulation is favoured by the AIC?