

# Exercises\_tips

Stefano Larsen

4/21/2021

## Exercises

This script describes some easy exercises related to the different topics presented in the other scripts. To run the exercises the libraries loaded in the scripts should be still available.

It is also expected that the objects created in the various scripts are still present.

Therefore the exercises and tips are also presented at the end of the related script.

## Exercises related to the *SpatCourse\_diversity.Rmd* script

```
library(vegan)
```

The students are invited to calculate Shannon and Simpson index on community matrices already available in R using both the *classic* and *Hill* approach. You could use data such as *data(dune)* or *data(BCI)*. These are datasets on meadow vegetation species cover (dune) or tree counts in the Barro Colorado Island (BCI). **In both datasets, rows=sites and cols=species.**

### TIP

to load a data object associated with e.g. the vegan package:

```
data(dune)
head(dune)
```

```
##   Achimill Agrostol Airaprae Alop geni Anthodor Bellpere Bromhord Chenalbu
## 1         1         0         0         0         0         0         0         0
## 2         3         0         0         2         0         3         4         0
## 3         0         4         0         7         0         2         0         0
## 4         0         8         0         2         0         2         3         0
## 5         2         0         0         0         4         2         2         0
## 6         2         0         0         0         3         0         0         0
##   Cirsarve Comapalu Eleopalu Elymrepe Empenigr Hyporadi Juncarti Juncbufo
## 1         0         0         0         4         0         0         0         0
## 2         0         0         0         4         0         0         0         0
```

```
## 3      0      0      0      4      0      0      0      0
## 4      2      0      0      4      0      0      0      0
## 5      0      0      0      4      0      0      0      0
## 6      0      0      0      0      0      0      0      0
##   Lolipere Planlanc Poaprat Poatriv Ranuflam Rumeacet Sagiprocc Salirepe
## 1         7         0         4         2         0         0         0         0
## 2         5         0         4         7         0         0         0         0
## 3         6         0         5         6         0         0         0         0
## 4         5         0         4         5         0         0         5         0
## 5         2         5         2         6         0         5         0         0
## 6         6         5         3         4         0         6         0         0
##   Scoraute Trifprat Trifrepe Vicilath Bracruta Callcusp
## 1         0         0         0         0         0         0
## 2         5         0         5         0         0         0
## 3         2         0         2         0         2         0
## 4         2         0         1         0         2         0
## 5         3         2         2         0         2         0
## 6         3         5         5         0         6         0
```

Then, create a dataframe that holds information on the number of species, and the diversity indices for each site. You can then *remove* 50% of the species and calculate again the diversity indices. Make note of the differences. What are the proportional changes in classic and Hill-based indices?

## TIP

to create a dataframe with e.g. number of species and shannon as columns using the dune data

```
my.df=data.frame(richness=specnumber(dune),
                 shannon=diversity(dune, 'shannon')
                 )
head(my.df)
```

```
##   richness  shannon
## 1         5 1.440482
## 2        10 2.252516
## 3        10 2.193749
## 4        13 2.426779
## 5        14 2.544421
## 6        11 2.345946
```

## TIP

to randomly remove 50% of the species from the dune data, use *sample*

```
half_dune=dune[,c(sample(1:30, 15))]
```

You can compare the mean values of each indices in the original and *halved* dataset using `colMeans()`

## Exercises related to the *SpatCourse\_spatial.Rmd* script

The key point of this exercise is to understand how to interpret a variogram. The description and tips are presented at the end of the script. In the *SpatCourse\_spatial.Rmd* script, you have simulated some spatially structured diversity data using the *ncf::rmvr.spa* function (script line:~132). Now you can simulate diversity data without spatial structure, and explore the variogram.

- What happens if you simulate *bird.div* ('div') without adding the spatially structured noise (i.e. adding 'rnorm ()' instead of 'div.ncf')?
- How does the variogram look like in this case?
- Would the use of spatial GLS model still be justified?

There are **TIPS** at the end of the *SpatCourse\_spatial.Rmd* script.

## Exercises related to the *SpatCourse\_spatial2.Rmd* script

Now that you are familiar with spatial data, variograms and GLS command, this exercise is a simple extension.

Variogram plots assume that the direction of autocorrelation is the same across all directions. This is called *isotropic* assumption. However, many real landscapes are non-isotropic (e.g. rivers, mountain ranges, etc).

- You are asked to use the landscapes simulated in the *SpatCourse\_spatial2.Rmd* script to plot variograms showing the patterns of autocorrelation over along different directions. **This is easy, look for the tips at the end of the *SpatCourse\_spatial2.Rmd* script.**

## Exercises related to the *SpatCourse\_SSN1.Rmd* script.

- The exercises related to this script focus on plotting variograms over the three distances of river networks (flow-connected, flow-unconnected and Euclidean). You can modify the scripts already used to plot variograms for other variables.
- Then you should also fit models for species richness. Start with a non-spatial model, and then a spatial models with all three autocovariance components. Compare then the models. **See tips at the end of the *SpatCourse\_SSN1.Rmd* script**

## Exercises related to the *SpatCourse\_SSN2.Rmd* script.

The focus of this exercise is simply to run another glmssn model using a different response variable from the available data. To explore different distributions, we can model the variable called *C16*, which is the number of days when the temperature was above 16C. This is a count variable that should be modelled using a ‘poisson’ distribution.

Again, there are TIPS at the end of the script.

- Fit a poisson model for the variable *C16* using again elevation and slope as covariates.
- Exclude non-significant predictors.
- Predict the *C16* over the prediction locations (e.g. Knapp stream) using your model.

Again, TIPS at the end of the *SpatCourse\_SSN2.Rmd* script.