

# Exercices on session *Tidyverse 2/2*

For all the following exercises, use functions from the tidyverse, and pipes, whenever possible.

## BCI tree count

You are going to work on data from trees in Barro Colorado Island. These data are available in the package *vegan*, and are called *BCI*.

- Load the data (don't forget to look at the metadata in the help of the package).
- Get an overview of the data and check their class.
- Transform the data to a tibble.

## Transform the dataset

- Add a column with the plot ID (from BCI\_1 to BCI\_...) (and check what you have done).
- Create a tibble with the data in a long format, to have one row per species and plot as follow:

```
# A tibble: 11,250 x 3
  plot_ID gen_sp      abund
  <chr>   <chr>      <int>
1 BCI_1  Abarema.macradenia      0
2 BCI_1  Vachellia.melanoceras    0
3 BCI_1  Acalypha.diversifolia    0
4 BCI_1  Acalypha.macrostachya    0
5 BCI_1  Adelia.triloba          0
6 BCI_1  Aegiphila.panamensis     0
```

```

7 BCI_1 Alchornea.costaricensis 2
8 BCI_1 Alchornea.latifolia 0
9 BCI_1 Alibertia.edulis 0
10 BCI_1 Allophylus.psilospermus 0
# i 11,240 more rows

```

- Check the dimension of this new table (try to do it using a test).
- We now want to add a column for the genus and species name.
  - Let's do it with the function *separate* (don't replace the initial data, store the result in a new object).

Warning: Expected 2 pieces. Additional pieces discarded in 11250 rows [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].

- We get a warning message... It is not very explicit... Let's have a look at the full species name. For that, display the species list. Can we separate the column *gen\_sp* based on the separator "." ? Why?
- We want to keep only the characters that are before the second dot (so the genus and the species, removing the variety and subspecies). Let's split the *gen\_sp* column by the separator ".". We want to get a matrix with a column by string separated by ".". **NB:** if you say that you want to split by ".", this will be interpreted as "any character", so you need to write "\\." instead.
- Let's keep only the two first columns of this matrix and bind these columns to the long dataset.

New names:

```

* `` -> `...4`
* `` -> `...5`

```

- Rename the two new columns to have this:

```

# A tibble: 11,250 x 5
  plot_ID gen_sp      abund genus species
  <chr>   <chr>    <int> <chr>   <chr>
1 BCI_1 Abarema.macradenia 0 Abarema macradenia
2 BCI_1 Vachellia.melanoceras 0 Vachellia melanoceras
3 BCI_1 Acalypha.diversifolia 0 Acalypha diversifolia
4 BCI_1 Acalypha.macrostachya 0 Acalypha macrostachya
5 BCI_1 Adelia.triloba 0 Adelia triloba

```

```

6 BCI_1    Aegiphila.panamensis      0 Aegiphila panamensis
7 BCI_1    Alchornea.costaricensis 2 Alchornea costaricensis
8 BCI_1    Alchornea.latifolia    0 Alchornea latifolia
9 BCI_1    Alibertia.edulis      0 Alibertia edulis
10 BCI_1   Allophylus.psilospermus 0 Allophylus psilospermus
# i 11,240 more rows

```

- Get rid of the rows with an abundance of zero.

## Add environmental data

- Let's now add some environmental data from the dataset *BCI.env*. Load it and transform it to a tibble.
- Add a column with the plot ID (from BCI\_1 to BCI\_...)
- Reorder the levels of *Habitat* from the one with the most plots to the one with the least plots. There is a function for this, it is called *fact\_infreq*.
- Add a column called *EnvHet\_cat* taking the following values:
  - *low* when the environmental heterogeneity is lower than 0.08
  - *medium* when the environmental heterogeneity is between 0.08 and 0.49
  - *high* when the environmental heterogeneity is higher than 0.49
- Join the variable *Habitat* and *EnvHet\_cat* to the long dataset of abundance (make sure you use the appropriate join function).

## For next time

- Install the following package from CRAN: *patchwork*, *plotly*