# Changes documentation

Jess

10/08/2021

This is a source document for changes made on the BCCVL scripts for implementation on the EcoCommons R package.

Functions and script common vocabulary follows Hadley Wickham's style guide (which is based of the Google style guide) http://adv-r.had.co.nz/Style.html.

---

**Downloading the package and understanding the folder structure**

You can install the package using the following command devtools::install_github("AtlasOfLivingAustralia/ecocommons") library(ecocommons)

As BIOMOD2 has a A LOT of dependencies, it can take a while... something to think in the future if we decide to make the package available to the public

After you download it, you will see 4 documents and 3 folders. README and DESCRIPTION explains a bit about the package. The NAMESPACE shows all the functions that are being exported for the package, hiding the internal functions. It also shows a list of the functions used from other packages.

The R folder contains all the functions. The man folder is generated from roxygen2 and shouldn't be changed by hand. The inst folder contains extra files. Including "variables", where are the dataset I used for internal test, "original_source_files", with the original BCCVL script and "dependency_management", with a script to test the amount of dependencies the R package is using.

---

**General Rules**

All the functions created within EcoCommons starts with "EC_". The reason is that some of the functions are really similar to existing ones, e.g. read_raster. The algorithm scripts that I am turning into functions starts with "modelling_". The "utilities_" are internal functions and subfunctions. Some are specific to geographical models, to maxent, etc

The "EcoCommons_source.R" is the script I am working on as the initial/example script to run everything. The idea is that you open it and follow the script to run the models, at least internally.

In the begging of each R document there is some information on the roxygen format. Please don't remove this part, but feel free to edit the information if needed. Please note that subfunctions that are only used in one function are nested above the main function.

---

**Step-by-step process**

1st STEP I turned functions that were in a big script into individual files and rename them

2nd STEP Created the package via devtools

3rd STEP Added roxygen2 documentation for all functions

4th STEP Improved comments and familiarized with the functions

5th STEP - Update packages when possible - Transform algorithm scripts into functions - Removed unused functions

6th STEP - Test algorithm scripts (internally) - Set up workflow document

NEXT STEPS - Test package - Recognize authorship of previously written R functions - check license of the R packages that are in use

FUNCTIONS THAT STILL NEED IMPROVEMENT - EC_performance_2d - EC_plot_VIP - EC_save_dismo_eval - EC_modelling_maxent

---

**Renaming Functions**

Functions were renamed

Functions that were deleted

---

**Synthax rules**

# Notation and Naming

- avoid weird separators
- give your files short and meaningful names
- file names should be unique
- VARIABLES all in lower case with words separated by dots; nouns; avoid single letters
- FUNCTIONS initial capital letters and no dots; verbs

# Comments

- descriptive names can help to avoid unnecessary comments
- comments should not state the obvious
- start comments with # and one space; short codes can be placed after the code, with two spaces, #, and then one space
- functions need special comments, with one sentence description of the function, a list of arguments with a description (including data type) and description of the return value

# Sytax

- always use <- when assigning names to objects and avoid using = for assignment " Even though this distinction doesn't matter for the majority of the time, it is a good habit to use <- as this can be used anywhere, whereas the operator = is only allowed at the top level. In addition = closely resembles ==, which is the logical operator for equals to"
- avoid long line length (~80 characters)
- place spaces around all binary operators (=, +, -, <-, ==, ! = )
- always put a space after a comma and never before
- curley braces opening should never go on its own line and should always be followed by a new line. A closing curly brace should always go on its own line, unless followed by else, which should be contained within outward facing curly braces >}else{. Indent the code within curly braces

---

**Ideas to implement in the future**

- Easy to see list of available algorithms
- Submit package to CRAN
- Add changelog (list of new functions) on README GitHub
- Logo?
- Add Author(s) per function, and include link to report bugs
- Change raster to terra