Alkuin Koenig

2023/06/07

V2

**Description:**

Compute HYSPLIT trajectories from Rstudio on a windows machine or MAC (linux not tested yet), making use of the excellent splitr library (<https://github.com/rich-iannone/splitr>). I implemented easy parallel computing via Rstudio jobs. I’m also giving examples on how to plot the trajectories, calculate footprints (in regular lat/lon coordinates as well as polar coordinates), etc.

----- INITIAL SETUP -----

**Step 1: Install all the dependencies for splitR and the libraries I use in my scripts.**

Note that there is some overlap between what splitr requires and what my scripts tend to use (green color). Try to have all the below libraries installed before installing splitr. (Reminder: in the Rstudio console type install.packages(“mypackage”).

|  |  |  |  |
| --- | --- | --- | --- |
| **For splitR** |  | **For my scripts** | **Comment** |
| downloader (>= 0.4), |  | lubridate | datetime utils |
| dplyr (>= 0.8.3), |  | data.table | Fast data crunching |
| leaflet (>= 2.0.2), |  | dplyr | Data crunching |
| lubridate (>= 1.7.4), |  | tidyr | Data crunching |
| magrittr, |  | rstudioapi | Parallelization via Rstudio jobs |
| readr (>= 1.3.1), |  | here | Managing relative paths |
| RCurl (>= 1.95), |  | ggplot2 | Best plotting library |
| scales (>= 1.0.0), |  | patchwork | Library for multipanel plots |
| tidyr (>= 0.8.3), |  | circular | Utils for polar coordinates |
| usethis (>= 1.5.1) |  | maps | Worldmap for plotting |

**Step 2: Get the splitr library and install it on your machine.**

Note that while splitr version 0.4 has its own DOI (10.5281/zenodo.49106), this version was published in 2016 and is probably outdated now. So you can do one of two things:

1. Install the package from the package archive files that I placed into the folder **./splitr\_library**. In Rstudio, go to: Tools -> Install Packages … -> go to “install from” -> select “Package Archive File”. Then select the right version depending on your operating system (only windows and mac provided at the moment).
2. Clone or download the latest splitr version (the whole splitr R project) from github (<https://github.com/rich-iannone/splitr>). Open the splitr R project in Rstudio and build the package on your own machine, directly from within Rstudio by typing devtools::build() into the console. You might have to install the R-package “devtools” first.

**In any case: Note that the splitr library comes with its own compiled version of the HYSPLIT code (for windows, linux, mac. Right version automatically detected), so you don’t need to install HYSPLIT on your machine separately!**

**Step 3: Open in Rstudio the R project file (HYSPLIT\_R\_scripts\_V1.Rproj)**

* **Important note:** Here and from now on, it is important to open all scripts from within the R project, if you want to work things “out of the box”. This is because of the way relative paths are defined! I’m using the library “here” for relative paths. here::here() simply points to the location of the R project file currently opened.
* to avoid trouble be sure to respect the folder structure of the project!
* In all cases, I mark the project file base folder with “**./**”. For example, “**./batch\_results/**” means “go to the folder where the R project is located, then from there into the folder batch\_results”

----- EXAMPLES -----

**Step 4: Run the first example to verify that everything is set up nicely**

1. Create the folder: **./batch\_results/Ex1\_AMS**. (if it already exists, erase it before)
2. We need to download the meteorological input (we’re using 2.5x2.5 ncar/ncep in this example). Open the script **“forcedownload\_meteofile.R”** (from within the main project!), set the variables as listed below, and run the script.

|  |
| --- |
| startdate = as.Date("2016-07-05") |
| enddate = as.Date("2016-07-15") |
| meteo\_output\_dir = paste0(here::here(),"/meteo\_input/met\_ncar\_ncep") |
| meteo\_type = "reanalysis" |

1. A download (~100 MB) from the ftp server: <ftp://arlftp.arlhq.noaa.gov/archives/reanalysis/> should initiate.   
   **IMPORTANT NOTE:** If the download does not work, it might be due to the settings of your internet connection. This manifests usually as a timeout. For example, when we tried, the download did NOT work from the **wifi** of our universityor **EDUROAM**, but it worked from an ethernet connection in the lab.
2. After the download finished, verify that you have the ncar/ncep meteorological input file (RP201607.gbl) in the right folder (./meteo\_input/met\_ncar\_ncep/)
3. **Verify that your R working space is empty. Empty if it’s not!!!** This is important, because your working space will be copied once for each job (this is how parallelization was implemented). Thus, it will clutter your memory if you have much too data in your working space already!
4. Now we can do the HYSPLIT runs. Open and run the example script: **run\_HYSPLIT\_batch\_Ex1.R.** Reminder: Make sure to open the script from within the Rproject.
5. Check on the status of the jobs (Note: In Rstudio, you can check on the status of jobs by clicking on “jobs” in the lower left corner, next to “Console” and “Terminal”). After all jobs have finished (note that some jobs may take longer than others!), check if the trajectories were created. In the folder **./batch\_results/Ex1\_AMS/.** there should be 4 individual \*.rds files in the folder, each weighting around ~30 kB.
6. Congrats! You just ran HYSPLIT from Rstudio….
7. To plot the trajectories you just ran, open the Rstudio notebook: **HYSPLIT\_plotting\_examples.rmd.** Adapt the input folder variable at the beginning ***path\_to\_files*** so that it points to “**Ex1\_AMS**”, and run the notebook.

**Step 5: Run the second example**

1. Create the folder: **./batch\_results/Ex2\_montblanc**. (If it already exists, erase it before)
2. This time we will use 1x1 GDAS1 as meteorological input. Open the script **“forcedownload\_meteofile.R”**, set the variables as listed below, and run the script.

|  |
| --- |
| startdate = as.Date("2016-07-05") |
| enddate = as.Date("2016-07-15") |
| meteo\_output\_dir = paste0(here::here(),"/meteo\_input/met\_gdas1") |
| meteo\_type = "gdas1" |

1. A chain of downloads from the ftp server: ftp://arlftp.arlhq.noaa.gov/archives/gdas1/gdas1.jun16.w5 should initiate. Note that these downloads are heavier than in the previous case and may take a while!
2. After all downloads have finished, verify that you have the following gdas1 meteorological input files in the right folder (./meteo\_input/met\_gdas/)
   1. gdas1.jun16.w4
   2. gdas1.jun16.w5
   3. gdas1.jul16.w1
   4. gdas1.jul16.w2
   5. gdas1.jul16.w3
   6. gdas1.jul16.w4
3. Now we can do the HYSPLIT runs. Open the example script: **run\_HYSPLIT\_batch\_Ex2.R** and run it (again, before running make sure that your working space is empty)!
4. After all the jobs have finished, check if the trajectories were created. There should be 24 individual .rds files in the folder: ./batch\_results/Ex2\_montblanc/. Each file should be around ~10 kB.
5. To plot the trajectories you just ran, open the Rstudio notebook: **HYSPLIT\_plotting\_examples.rmd.** Adapt the input folder variable *path\_to\_files* so that it points to “Ex2\_montblanc”, and run the notebook.

**Step 6: (Optional) Check out the other example scripts!**

For example, there is an example script on how to set up a matrix run (trajectories not emitted from one single point, but from a matrix of points)

----- DESIGN YOUR OWN RUNS -----

**Step 7: Adapt the scripts to your needs**

1. Before doing any parallel runs, use the script **forcedownload\_meteofile.R** to download the meteorological input you need. If this automated script doesn’t work, got to NOAAs ftp (<ftp://arlftp.arlhq.noaa.gov/archives>) and download it manually. There you can also get gfs0.25 as input.
2. Make a copy of one of the example scripts and rename it
3. Change the input variables to your needs. More infos are found as comments in the scripts!
4. Note that you can change the numbers of individual jobs (variable name: jobnumber), this defines on how many processes you want to split the HYSPLIT run (parallel computing). My, admittedly powerful, working laptop does 10 without issues. I’m confident that even the most potato laptop should be able to do 4.
5. If you’re finished with the edits, make sure the Rstudio working directory is empty, run the script and wait for all processes to finish (depending on how extensive your run is, this can take a while!)
6. You can always plot your output using the notebook **HYSPLIT\_plotting\_examples.rmd** (don’t forget to change the input folder variable *path\_to\_files* accordingly). Most of the time you won’t be interested in all the plots, or want a different plot. The notebook only gives some basic examples, feel free to adjust.

**Additional notes and debugging:**

In this section, I collect unsorted information about problems a user could encounter during setup and use of the scripts.

* The trajectory run can fail if you have special characters in one of your paths (to the meteorological input, the HYSPLIT executable, the output folder, etc.)! For example, things may not work out if you have a folder or file called “thèse” or “Äthiopien” or “Coruña” or “my folder” (note the space) or “food&drinks” or whatever. Be clean with your folder and file names!
* If you use the option extended\_met = TRUE (i.e. you want additional meteorological information to be included in the trajectory output), the correct fields (e.g. Temperature, mixing height, relative humidity) must be included in your input meteorological file, otherwise the run will fail! These fields are usually already included in gdas1 and reanalysis, but if you are using ERA5 for example then it depends on how the conversion from ERA5 to ARL was made! If your run fails with extended\_met = TRUE, your first attempt for debugging should be to set extended\_met = FALSE.