POLARIS — Import for PSM comparisons

D G Rossiter d.g.rossiter@cornell.edu

04-July-2021

Contents

Introduction

POLARIS is the result of harmonizing diverse SSURGO and STATSGO polygon data with the DSMART algorithm to produce a raster soil series map (30 m resolution) and then extracting property information from Official Series Descriptions.

It is explained in: Chaney, N. W., Minasny, B., Herman, J. D., Nauman, T. W., Brungard, C. W., Morgan, C. L. S., McBratney, A. B., Wood, E. F., & Yimam, Y. (2019). POLARIS soil properties: 30-m probabilistic maps of soil properties over the contiguous United States. Water Resources Research, 55(4), 2916–2938. Scopus. https://doi.org/10.1029/2018WR022797

POLARIS is organized as $1 \times 1^\circ$ tiles.

To use this script:

1. Ajust the directory structure to your system
2. Select a property and quantile and select a depth slice, using the YAML header or by knitting with parameters.
3. Select an Area of Interest, using the YAML header or by knitting with parameters.
4. Either compile to HTML or PDF ("knit"), or "Run All" within R Markdown.
5. The processed tile will be in the directory structure, in a subdirectory named for the AOI.

This GeoTIFF can then be read into R and compared with other PSM products.

Directories

Set base directories, specific to the local file system.

Directory `base.dir.polaris.import` is where downloaded large (up to 35 Mb) GeoTIFF are located. Because of their size they may be on a separate file system, e.g., removable or networked drive.

```
base.dir.import <- "/Volumes/Pythagoras/ds/DSM_import/"
base.dir.polaris.import <- paste0(base.dir.import, "POLARIS/")
```

These are the base of destination directories built below.

Packages

```
library(terra)  # for raster import and display
```

Parameters

Parameters for this run:

```
print(paste("lrc_long:", params$lrc_long, "; lrc_lat:", params$lrc_lat, "; size:", params$size))
```

```
[1] "lrc_long: -86 ; lrc_lat: 38 ; size: 1"
```

```
print(paste("voi.n:", params$voi.n, "; depth.n:", params$depth.n))
```

```
[1] "voi.n: 6 ; depth.n: 1"
```

```
print(paste("quantile.n:", params$quantile.n))
```

```
[1] "quantile.n: 4"
```

Variable and quantile of interest

Define the variables for the soil property and layer of interest.

- `p5` - 5% quantile from the machine-learning methods;
- `p50` - median of the distribution;
- `p95` - 95% quantile;
- `mean` - mean of the distribution.

```
quantile.list.polaris <- c("p5", "p50", "p95", "mean")
```

The list of properties and their units of measure is here.

Relevant here are:

- clay - clay percentage, %
- silt - silt percentage, %
- sand - sand percentage, %
- ph - soil pH in H2O, N/A
- om - organic matter, log10(%)
- bd - bulk density, g/cm3

There are also soil hydrological parameters we do not process.

```
voi.list.polaris <- c("clay", "silt", "sand", "ph", "", "om", "bd", "")
```

Set the property and quantile from the YAML or rendering parameters:

```
voi.polaris <- voi.list.polaris[params$voi.n]
quantile.polaris <- quantile.list.polaris[params$quantile.n]
```

Depth slice

Depth slices, in cm:

```
depth.list.polaris <- c("0_5", "5_15", "15_30", "30_60", "60_100", "100_200")
```

Set the depth from the YAML or rendering parameters. Then build a full layer name with the property, depth, and quantile:

```
depth.polaris <- depth.list.polaris[params$depth.n]
(voi_layer <- paste(voi.polaris, depth.polaris, quantile.polaris, sep="_")) # layer of interest
```

```
[1] "om_0_5_mean"
```

Area of Interest (AOI)

POLARIS data is served in $1 \times 1°$ tiles using WGS84 geographic coordinates. Specify the *lower-right corner*, then compute the upper-right corner 1° west and north.

Specify the lower-right corner from the YAML or rendering parameters:

```
tile.lrc <- c(params$lrc_long, params$lrc_lat) # lower-right corner
```

Compute the upper-left corner:

```
tile.ulc <- c(tile.lrc[1]-1, tile.lrc[2]+1) # upper-left corner
```

A prefix for directories, to keep AOI results separate.

```
AOI.dir.prefix <- paste0("lat", tile.lrc[2], tile.ulc[2],
                         "_lon", tile.ulc[1], tile.lrc[1])
```

Import the POLARIS tile

POLARIS tiles as GeoTIFFs can be downloaded here. These are organized as hierarchical directory: property, depth, quantile. The file name contains the tile bounding box, e.g. `lat4142_lon-78-77.tif`.

Download the file if we don't already have it.

Second, make sure the directory exists, creating it if necessary.

```
(dest.dir.polaris.import <-  paste0(base.dir.polaris.import,
                            AOI.dir.prefix, "/",
                            voi.polaris, "/",
                            quantile.polaris, "/",
                            depth.polaris))
```

```
[1] "/Volumes/Pythagoras/ds/DSM_import/POLARIS/lat3839_lon-87-86/om/mean/0_5"
```

```
if (!dir.exists(dest.dir.polaris.import)) {
   dir.create(dest.dir.polaris.import, recursive = TRUE)
}
```

Finally, download the tile.

```
polaris.tile <- paste0("lat", tile.lrc[2], tile.ulc[2],
                       "_lon", tile.ulc[1], tile.lrc[1],
                       ".tif")
(dest.file <- paste0(dest.dir.polaris.import, "/", polaris.tile))
```

```
[1] "/Volumes/Pythagoras/ds/DSM_import/POLARIS/lat3839_lon-87-86/om/mean/0_5/lat3839_lon-87-86.tif"
```

```
if (!file.exists(dest.file)) {
   download.file(
      url =
         paste0("http://hydrology.cee.duke.edu/POLARIS/PROPERTIES/v1.0/",
                voi.polaris, "/", quantile.polaris, "/", depth.polaris, "/",
                polaris.tile),
      destfile = dest.file,
      method = "auto")
} else {
   print("Local copy of file already exists")
}
```

```
[1] "Local copy of file already exists"
```

These are quite large (8 – 35 Mb) and therefore are read into the import location. They can be deleted after they have been read into R.

Check

Import to R and print/plot, to check:

```
r <- terra::rast(dest.file)
print(r)
```

```
class       : SpatRaster
dimensions  : 3600, 3600, 1  (nrow, ncol, nlyr)
resolution  : 0.0002777778, 0.0002777778  (x, y)
extent      : -87, -86, 38, 39  (xmin, xmax, ymin, ymax)
coord. ref. : +proj=longlat +datum=WGS84 +no_defs
source      : lat3839_lon-87-86.tif
name        : lat3839_lon-87-86
```

```
plot(r)
```

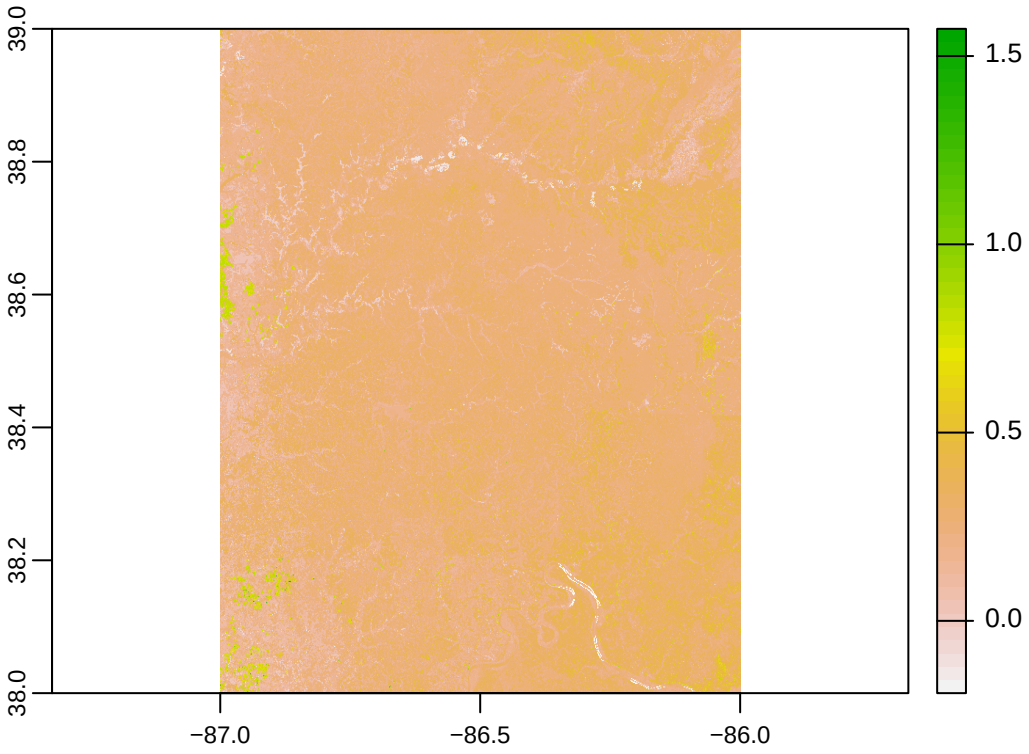

Figure 1: Checking the imported POLARIS tile