

potentialResourcesNT_DataPrep

Tati Micheletti

27 June 2022

Overview

This is a data preparation module to harmonize different anthropogenic disturbance datasets, more specifically, mining and oil/gas. It's intended for the Northwest Territories region (default) and is idiosyncratic. This means this module is NOT generalizable, but can be used as basis for other types of development. The objective is to create one standardized layer for each of the potential resources that has increasing values for most prioritized places (i.e., higher values, more likely structures will appear).

Usage

```
if(!require("Require")){
  install.packages("Require")
}
library("Require")
Require("googledrive")
Require("SpaDES.core")

# Pass your email for authentication (used for non-interactive calls)
googledrive::drive_auth(email = "tati.micheletti@gmail.com")
options(reproducibile.useTerra = FALSE) # Workaround while reproducibile is not yet fully functional with

# If you load the project, set the directory where your modules are located
moduleDir <- dirname(getwd())

setPaths(modulePath = moduleDir,
  cachePath = checkPath(file.path(getwd(), "cache"),
    create = TRUE),
  outputPath = checkPath(file.path(getwd(), "outputs"),
    create = TRUE),
  inputPath = checkPath(file.path(getwd(), "inputs"),
    create = TRUE),
  rasterPath = checkPath(file.path(getwd(), "temp_raster"),
    create = TRUE))

getPaths() # shows where the 4 relevant paths are

times <- list(start = 2011, end = 2011)
```

```

parameters <- list(
  #.progress = list(type = "text", interval = 1), # for a progress bar
  # Default values, don't need to be passed but are here as examples
  potentialResourcesNT_DataPrep = list(whatToCombine = data.table::data.table(datasetName = c("oilGas",
                                                                                      "mining"),
                                     dataClasses = c("potentialOilGas", "potentialOilGas",
                                                                                      "potentialMining", "potentialMining"),
                                     toDifferentiate = c(NA, "C2H4_BCR6_NT1",
                                                                                      "CLAIM_STAT", "PERMIT_STA"),
                                     activeProcess = c(NA, NA,
                                                                                      "CLAIM_STAT", "PERMIT_STA")))
)
modules <- list("potentialResourcesNT_DataPrep")
objects <- list()
inputs <- list()
outputs <- list()

disturbanceList <- simInitAndSpades(times = times,
                                     params = parameters,
                                     modules = modules,
                                     objects = objects)

```

Parameters

This module has one parameter that can be adjusted by the user:

- whatToCombine: This is a data.table and defaults to:

```

DT <- data.table::data.table(datasetName = c("oilGas", "oilGas", "mining", "mining"),
                             dataClasses = c("potentialOilGas", "potentialOilGas",
                                                                 "potentialMining", "potentialMining"),
                             toDifferentiate = c(NA, "C2H4_BCR6_NT1", "CLAIM_STAT", "PERMIT_STA"),
                             activeProcess = c(NA, NA, "CLAIM_STAT", "PERMIT_STA"))

print(DT)

```

##	datasetName	dataClasses	toDifferentiate	activeProcess
## 1:	oilGas	potentialOilGas	<NA>	<NA>
## 2:	oilGas	potentialOilGas	C2H4_BCR6_NT1	<NA>
## 3:	mining	potentialMining	CLAIM_STAT	CLAIM_STAT
## 4:	mining	potentialMining	PERMIT_STA	PERMIT_STA

Here the user should specify a data.table with the `dataName` and `dataClasses` from the object `disturbanceList` from the module `anthroDisturbance_DataPrep` to be combined. The table also contains a column identifying which to be used to filter active processes for mining (i.e., `CLAIM_STAT` and `PERMIT_STA`). For Oil/Gas, it needs to identify which layer is the potential one (`C2H4_BCR6_NT1`) and which is used to constrain where oil and gas will be added. For oil and gas, the other potential layer (exploration permits) is used as a starting point to add structures, followed by randomly placing them in the highest values of `C2H4_BCR6_NT1` and going down until the total amount is reached. For mining, `CLAIM_STAT` is the potential exploration, while `PERMIT_STA` are the ones that might become CLAIMS. The most likely values are CLAIMS and followed by PERMITS;

Events

This module contains four events, which are similar in objective, but specific in the inputs and outputs. All events (`createPotentialMining`, `createPotentialOilGas`, `createPotentialCutblocks`, and `replaceInDisturbanceList`) aim at working on all current and potential disturbance layers, unifying these, and setting the highest values as the locations that need to be filled with new developments first, or prepare potential layers. An important note is that for the function `makePotentialCutblocks()`, we recreated the steps done by ENR (J. Hodson) to create the potential forest. Specific information can be found as comments in the function.

Data dependencies

Defaults

This module can be run without any inputs from the user and will automatically default to an example in the Northwest Territories of Canada (i.e., union of BCR6 and NT1).

Input data

The module expects four inputs:

- **disturbanceList**: List (general category or sector) of lists (specific class or sub-sector) needed for generating disturbances. The sub-sector last list contains:
 - + Outer list's names: **dataName** from **disturbanceDT**;
 - + Inner list's names: **dataClass** from **disturbanceDT**, which is a unique class after harmonizing, except for any potential resources that need idiosyncratic processing. This means that each combination of **dataName** and **dataClass** (except for 'potential') will only have only one element;
- **historicalFires**: List per YEAR of burned polygons. It needs to contain at least the following columns: YEAR or DECADE. The default layer was created by ENR for the NT.

The other two inputs needed by this module are the the study area (**studyArea**) and a raster (**rasterToMatch**) that matches the study area and provides the spatial resolution for the simulations.

```
df_inputs <- SpaDES.core::moduleInputs("potentialMiningAndOil", moduleDir)
knitr::kable(df_inputs)
```

Output data

The module outputs only one object, the **disturbanceList**, which is a modified version of the input one, where multiple potential layers (i.e., mining and oilGas) have been replaced by only one layer, with the highest values being the ones that need to be filled with new developments first.

```
df_outputs <- SpaDES.core::moduleOutputs("potentialMiningAndOil", moduleDir)
knitr::kable(df_outputs)
```

Links to other modules

This module can be combined primarily with `anthroDisturbance_DataPrep` and `anthroDisturbance_Generator`, both modules being generic and applicable in other contexts, potentially without modifications. Such

module collection can also be used in combination with (a) landscape simulation module(s) (i.e., LandR) and caribou modules (i.e., caribouRSF, caribouRSF_NT, caribouPopGrowth) to improve realism on simulated landscape and caribou RSF and population growth forecasts.