Introduction

Due to the loss of support for both raster and the older version of sp, changes were required to avoid deprecated packages being used in existing software that relied on those 2 packages. However, due to the nature of how intertwined R packages can be, as well as the reliance of many packages on sp and raster, this process can be more complicated than what can be found at first glance. Thankfully, packages like terra and sf have very similar functionality to packages like raster and sp, and with the help of those newer packages, it becomes a much smoother process to transition.

However, this process did not remain without some roadblocks and difficulties, especially in the cases where function calls to sp and raster functionality is deeply embedded within a large program, other packages that have dependencies on sp and raster are used, and especially when data types from sp and raster and utilized throughout an application.

These changes will be discussed in the context of switching the Spatial Ecology Gateway software over from raster/sp to terra/sf. The Spatial Ecology Gateway also utilized the move, adehabitatHR, and mkde packages, which all relied upon raster and/or sp. The Spatial Ecology Gateway also utilized datatypes from the sp and raster packages, such as the SpatialLinesDataFrame and similar data types from sp, as well as the RasterLayer (and was set up to use RasterStack as well) from the raster package. Moreover, since the functionality involved using said datatypes and functions from said packages to generate visualizations that would be utilized throughout the program, the functionality and dependencies were often intertwined.

The purpose of this is firstly to document what changes were made and what problems were solved on the way to do so. It is meant to give some examples of how I was able to change the code to be able to maintain functionality within an application while changing the code to be able to use the newer and more supported packages. I will summarize the changes below.

Raster/MKDE Function Changes

The MKDE package contained several functions that were dependent upon at least one of sp or raster. This package is publicly available on CRAN and on GitHub and so in finding the functions that utilized sp and raster, I could test their functionality and compare their output to my new alternative functions that instead utilized sf and terra. Some of the functions include setMinimumZfromRaster, setMaximumZfromRaster, writeRasterToXDMF, writeRasterToVTK, and the most important function out of these mkdeToRaster.

(Ask Bob for permission (because of GPL-3.0 license) to add side-by-side code comparison pictures here with descriptions line-by-line)

The raster package is one of the packages that was losing support, however some of the functionality from the raster package was directly within the Spatial Ecology Gateway. Again, this package is also publicly available on CRAN and GitHub and so I took a look at functions that were being used including rasterToContour.

(Ask Bob for permission (because of GPL-3.0 license) to add side-by-side code comparison pictures here with descriptions line-by-line)

Data Type Conversions

One of the largest changes within the conversion from sp to sf and raster to terra are the datatype that are associated with sp and raster. The fact that the data types used within these packages are directly from those two packages means that not only do existing functions have to be replaced, the format and handling of the data will be different, both in the process leading up to the creation of these objects with the mentioned data types but also in the process after the creation of those data types, where the data that is returned by those functions will have to reused many other times throughout the program. This raises a question about the formatting of these data types, as in many cases the formatting of the sf objects versus the formatting of something like a SpatialLinesDataFrame caused issues that needed to be handled.