Evening Hade, Lab 2, Part 1, Step 3

Creating a Space Time Cube

For GIS 5571

25 October 2024

```
#begin with imports
import os
import requests
import arcpy
import zipfile

#set up arcpro project, map, and coordinate system
project = arcpy.mp.ArcGISProject("CURRENT")
map = project.listMaps()[0]
ref = arcpy.SpatialReference(4326) #using WGS 1983

#set up output path where everything is saved
outputPath = r"C:\ArcGIS\Projects\Lab2_Part1\PRISM"
#but i only want to make it if it does not already exist
if not os.path.exists(outputPath):
    os.makedirs(outputPath)
    print("Created output directory: ", outputPath)
else:
    print("Output directory is already assigned to: ", outputPath)
```

```
Output directory is already assigned to:  C:\ArcGIS\Projects\
Lab2_Part1\PRISM
```

```
#base url we will add to
baseUrl = 
'https://services.nacse.org/prism/data/public/normals/4km/ppt/'

#for loop to add the correct endpoint to the url and download all the
data
for month in range(1, 13):
    url = f"{baseUrl}{month}" #format variable as string (cool fancy
trick 10/10)
    zipPath = os.path.join(outputPath, f"PRISM_ppt_month{month}.zip")

    #download ZIP file if it hasn't been downloaded
    if not os.path.exists(zipPath):
        response = requests.get(url)
        if response.status_code == 200: #success
            with open(zipPath, 'wb') as zipFile:
                zipFile.write(response.content)
            print(f"Downloaded ZIP file for month {month} to: ", 
zipPath)
```

```
        else: #error
            print(f"Failed to download ZIP file for month {month}.
Status code: ", response.status_code)
    else:
        print(f"ZIP file for month {month} already downloaded.")
```

```
ZIP file for month 1 already downloaded.
ZIP file for month 2 already downloaded.
ZIP file for month 3 already downloaded.
ZIP file for month 4 already downloaded.
ZIP file for month 5 already downloaded.
ZIP file for month 6 already downloaded.
ZIP file for month 7 already downloaded.
ZIP file for month 8 already downloaded.
ZIP file for month 9 already downloaded.
ZIP file for month 10 already downloaded.
ZIP file for month 11 already downloaded.
ZIP file for month 12 already downloaded.
```

```
#ONLY RUN ME ONCE

#unzip files
    with zipfile.ZipFile(zip_path, 'r') as zip_ref:
        zip_ref.extractall(outputPath)
    print(f"Extracted files for month {month} to: ", outputPath)

#DO NOT RUN IF ALREADY ADDED TO MAP

#create gdb to save mosaic to if it doesn't exist
gdbPath = os.path.join(outputPath, "PRISM_MosaicDataset.gdb")
if not arcpy.Exists(gdbPath):
    arcpy.CreateFileGDB_management(outputPath,
"PRISM_MosaicDataset.gdb")
    print("Created geodatabase:", gdbPath)
else:
    print("GDB already exists at: ", gdbPath)

#create and save mosaic dataset if it doesn't exist
mosaicPath = os.path.join(outputPath, "PRISM_MosaicDataset.gdb",
"PRISM_Mosaic")
if not arcpy.Exists(mosaicPath):
    arcpy.CreateMosaicDataset_management(gdbPath, "PRISM_Mosaic", ref)
    print("Created mosaic dataset:", mosaicPath)
else:
    print("Mosaic dataset already exists at: ", mosaicPath)

#add BILs to the mosaic dataset
bil = [os.path.join(outputPath, f) for f in os.listdir(outputPath) if
f.endswith(".bil")]
```

```
arcpy.management.AddRastersToMosaicDataset(mosaicPath, "Raster
Dataset", bil)
```

```
GDB already exists at:  C:\ArcGIS\Projects\Lab2_Part1\PRISM\
PRISM_MosaicDataset.gdb
Mosaic dataset already exists at:  C:\ArcGIS\Projects\Lab2_Part1\
PRISM\PRISM_MosaicDataset.gdb\PRISM_Mosaic
```

```
<Result 'C:\\ArcGIS\\Projects\\Lab2_Part1\\PRISM\\
PRISM_MosaicDataset.gdb\\PRISM_Mosaic'>
```

```
if not arcpy.ListFields(mosaicPath, "StdTime"): #apparently "StdTime"
is a standard variable name. i tried "Month" and it didn't work. but
that also may have been unrelated
    arcpy.AddField_management(mosaicPath, "StdTime", "DATE")
    print("Added StdTime field to mosaic dataset.")

#OBJECTID aligns with month order because the files appear
alphabetically in the folder and were added in that order to the
mosaic
#so i will determine the month of the year based on the OBJECTID
with arcpy.da.UpdateCursor(mosaicPath, ["OBJECTID", "StdTime"]) as
cursor:
    for row in cursor:
        month = row[0]
        row[1] = f"2024-{month:02d}-01"  #set date as first day of
month
        cursor.updateRow(row)
print("Populated StdTime field.")

#for some reason i get an error message but it also worked?
#maybe it worked in a previous code iteration and i didn't notice?
#anyway, i am too nervous to change it
```

```
----------------------------------------------------------------------
-----
RuntimeError                              Traceback (most recent call
last)
In  [10]:
Line 11:    cursor.updateRow(row)

RuntimeError: The value type is incompatible with the field type.
[StdTime]
----------------------------------------------------------------------
-----
```

```
#define time dimension so our raster has multiple dimensions. you
could say it is multidimensional, even.
arcpy.md.BuildMultidimensionalInfo(
    in_mosaic_dataset=mosaicPath,
    variable_field="Variable",
    dimension_fields="StdTime", #define month as the time dimension
    delete_multidimensional_info="NO_DELETE_MULTIDIMENSIONAL_INFO"
)

#make multidimensional raster layer
arcpy.md.MakeMultidimensionalRasterLayer(
    in_multidimensional_raster=mosaicPath,
    out_multidimensional_raster_layer="PRISM_Mosaic_Multi",
    variables="Dataset",
    dimension_def="ALL",
    dimension="StdTime" #reset month as time dimension
)

#these functions are both copied from the Esri website
```

```
<Result 'PRISM_Mosaic_Multi'>
```

```
#ONLY RUN ME ONCE

#where i will export my space time cube
cubePath = os.path.join(outputPath, "PRISM_SpaceTimeCube")
#now make the space time cube...
#or try
#it spent days crashing for me before i finally got it
# :P
try: #success
    arcpy.stpm.CreateSpaceTimeCubeMDRasterLayer(
        in_md_raster="PRISM_Mosaic_Multi",
        output_cube=cubePath,
        fill_empty_bins="ZERO"
    )
    print("Space-time cube created successfully at: ", cubePath)
except Exception as e: #error
    print("Error creating space-time cube: ", e)

#DO NOT RUN IT W*I*L*L* BRICK YOUR COMPUTER
```

```
Space-time cube created successfully at:  C:\ArcGIS\Projects\
Lab2_Part1\PRISM\PRISM_SpaceTimeCube
```