Ocean DB

Justin Nelson, Matthew Eberhard, Derek Hackenbracht, Will Hillhouse

CS4624 - Multimedia, Hypertext, and Information Access
Professor Edward Fox
Virginia Tech, Blacksburg VA 24061
April 18, 2023
Outline

1. Project Overview
2. Deliverables
3. General Approach
4. Timeline
5. Work Completed
6. Remaining Work
Project Overview

- **Our client:** Dr. Luis Escobar in the Department of Fish and Wildlife Conservation
- **His Goal:** Study the effects of climate change on the features of the ocean and the likely effects on human health and biodiversity.
- **Our Goal:** Collect relevant ocean condition data as far back as possible to help our client with his research.
Deliverables

- Collecting annual ocean salinity and surface temperature data dating as far back as possible in a GEOTiff file format.
- Additionally, we will write a lightweight script that will allow Dr. Escobar to pull new data in the future and update his database of files.
General Approach

- Already received data sources from the client
  - Matter of collecting and organizing the necessary data
- Build an understanding of the API for the data source
- Develop script for grabbing data between user-specified dates
Timeline

Feb 14 - Clarify Project Details with Client

Feb 27 - Local Download of Temperature and Salinity Data for Past 5 Decades

March 1 - Transfer Local Downloads to Client’s Folder

March 3 - Collect Earlier Data

March 5 - Basic API Calls in Sample Script

March 24 - Full API Functionality

March 31 - Ease of Use Features and Data Organization

April 14 - Implement Data Input Testing System
Old Conversion Technique - Raw Data Downloaded

- Downloaded historical data
- Data comes in as .nc file
- Must extract desired parameter and convert to TIFF format
Old Conversion Technique - Scripting for Data Transforms

```python
# open data set
nc_file = xr.open_dataset(folder_dir_str + "/" + filename)
# print(nc_file.data_vars) # prints the data variables available for extraction in .nc file

sT = nc_file['sea_surface_temperature'] # extract desired data
sT = sT.reindex(lat=list(reversed(sT.lat))) # reindex the latitude to flip the image
sT = sT.rio.set_spatial_dims(x_dim='lon', y_dim='lat') # set spatial parameters for new file
sT.rio.crs

sT.rio.write_crs("epsg:4326", inplace=True) # Define the CRS projection
sT.rio.to_raster("finalData/SurfaceTemp/" + date_time_obj.strftime('%Y') + "/" + tiff_file_name + ".tiff") # final tiff
```
Old Conversion Technique - Final Data Sample

Before

After
New Conversion Technique - Raw Data Downloaded

<table>
<thead>
<tr>
<th>Name</th>
<th>Long Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>sosaline_control_monthly_highres_2D_*</td>
<td>Monthly mean 2D Sea Surface fields</td>
<td>Local File</td>
</tr>
<tr>
<td>nav_lat</td>
<td>Latitude</td>
<td>Geo2D</td>
</tr>
<tr>
<td>nav_lon</td>
<td>Longitude</td>
<td>Geo2D</td>
</tr>
<tr>
<td>sosaline</td>
<td>Sea Surface Salinity</td>
<td>Geo2D</td>
</tr>
<tr>
<td>time_counter</td>
<td>time</td>
<td>—</td>
</tr>
</tbody>
</table>
New Conversion Technique - Scripting for Data Transforms
New Conversion Technique - Final Data Sample
Work Completed - Scripting

- Have base code for conversion technique
- Has been extended to extract data directly from API
- Specify months of data to grab, use same conversion techniques to convert new data

```python
import cdsapi

c = cdsapi.Client()

c.retrieve(
    'satellite-sea-surface-temperature',
    {
        'version': '2.1',
        'variable': 'all',
        'format': 'zip',
        'sensor_on_satellite': 'avhrr_on_nosa_19',
        'month': ['01', '02', '03', '04', '05', '06', '07', '08', '09', '10', '11', '12'],
        'day': ['01', '02', '03', '04', '05', '06', '07', '08', '09', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20', '21', '22', '23', '24', '25', '26', '27', '28', '29', '30', '31'],
    },
    'download.zip')
```
Work Completed - Testing

- Edge case testing for user input
  - Ensuring input is a valid date-time
  - Ensuring data availability for supplied date
- API calls
  - Tests for catching errors returned by API
  - Sending improper API calls
- File conversion testing
  - Testing faulty files (aka files with improper variable names, headers, etc.)
Remaining Work

- All assigned work has been completed
- Possible Improvements Include:
  - Frontend ease of use features for the script
  - Extending work to more sources
  - Machine learning model to analyze temperature and salinity data
Acknowledgements

- Dr. Luis Escobar - Virginia Tech (Virginia Polytechnic Institute and State University) | VT · Department of Fish and Wildlife Conservation
- Professor Edward A. Fox
- GTA Ryan Wood
References

- https://esgf-node.llnl.gov/search/cmip6/