Challenge - Environmental Modeller & Researcher (all gender)

We are excited to present you our challenge for the Environmental Modeller & Researcher position at Climate Farmers. Your willingness to invest your time in this process means a lot to us, and we sincerely appreciate your participation.

We eagerly anticipate the opportunity to delve into your outcomes together and use this as a chance to further understand your thinking and how you approach new tasks.

This challenge not only allows us to assess your skills but also provides you with valuable insights into the tasks that await you at Climate Farmers. It's a chance to discover if this role aligns with your expectations and can bring you both personal and professional fulfillment.

Thank you once again for embracing this challenge with us.

Good luck!

Introduction

The following challenge involves the creation of scripts to analyze gridded data products.

Challenge deliverables

You can present results in the format of your choice, such as a Jupyter or R Notebook, slides or a simple document including figure and code. Please include:

- A succinct description of the analysis steps performed
- A script supporting your approach
- Visualizations of results

Your script should be in R or Python and properly commented. Please include functions.

Preparation time should be more or less 1 day, depending on previous skills and knowledge.

If you get stuck at a given step, continue with something you are more comfortable with and feel free to use an alternative approach to attain similar results.

Please share a link to GitHub or a similar platform where you have uploaded your code (we thus check for basic git knowledge).

Challenge

The work involves the use and analysis of gridded data (usually remote sensing or model products), including land cover, climate and soil. You will need to download and visualize relationships between these over a chosen region.

Step 1: Region Selection

Decide on a specific region to work with. This region should be large enough to include various land use types but small enough to keep computation times reasonable. A country or region within a country for example.
The region can be defined using simple longitude and latitude limits or using a geospatial file (e.g. shape file or geojson). Use this region to filter downloaded data in the next steps.

Step 2: Data Acquisition

Download data for your region for:

- Monthly evapotranspiration, temperature, and precipitation covering the years 2000 to 2022 from (ERA5-Land): https://cds.climate.copernicus.eu/#!/home
- Land cover classification maps for the year 2020 from Land Cover Classes (Gridded Map):: https://cds.climate.copernicus.eu/#!/home
- Soil organic carbon (SOC) stock data: https://soilgrids.org/

If there are any questions about what exact variables should be used, use your own judgment to make a choice.

Step 3: Data Integration

Reproject the different rasters to match the one with lowest resolution, thus allowing joint analysis.

Step 4: Analysis and Visualization

Visualize the time series of climate and soil variables in the format of your choice, aggregated separately for different land cover classes.

Step 5: Sampling Design

Imagine you wanted to detect changes in SOC in the region in response to farm management practices or some other factor. so you need a sampling plan. Based on the SOC data you obtained, determine the number of samples that would need to be taken in the region in order to have a 95% confidence interval equal or less than 10% of the mean value. Assume a gaussian distribution.

Step 6: Free Task

This part is optional. It can involve any other type of visualization, modeling, statistical analysis or functionality of your choice. Some options are:

- The application of a soil carbon model (e.g. using the R package soilR) and plotting of results for a single point using your downloaded and/or artificial data as input.
- Your code translated from R to Python or vice-versa
- A more sophisticated sampling design, such as using stratified sampling
- Any additional functionality or automation in your code
- Any additional statistical analysis

Have fun!!