

Module 1: Climate API

Task description

API Meteo: <https://open-meteo.com/en/docs/climate-api>

Creas un script que se conecta a la API Meteo, saca los datos de `temperature_2m_mean`, `precipitation_sum` y `soil_moisture_0_to_10cm_mean` para Madrid (latitude=40.4165, longitude=-3.7026), Londres (latitude=51.507351, longitude=-0.127758) y Rio de Janeiro (latitude=-22.906847, longitude=-43.172896) desde 1950 hasta 2050, y representa en un gráfico la evolución de esas variables en el tiempo. Para la representación presenta el promedio + dispersión de los distintos modelos climáticos disponibles en la API. Ten en cuenta que probablemente necesitarás reducir la resolución temporal (la API devuelve con resolución diaria) para representar los datos de forma significativa.

Componentes de la solución:

Crear una Pull Request (PR) en vuestro repositorio personal en la que resolvais la tarea propuesta. En la descripción de la PR deberás añadir una o varias imágenes con los resultados obtenidos así como una breve descripción con conclusiones o comentarios añadidos.

En cuánto al código en sí mismo se espera que generes:

1. Una función `get_data_meteo_api` que llame a la API Meteo y saque los datos arriba descritos para cada ciudad.
 - a. La ciudad deberá definirse como un parámetro de entrada de la función `get_data_meteo_api` para que esta sea reutilizable para múltiples llamadas.
 - b. Tendría sentido crear una función auxiliar genérica para hacer llamadas a APIs. Recuerda que cuando se trabaja con APIs hay que gestionar el potencial rate limit (cool off), el status code que devuelve la API y los potenciales errores que puedan surgir, para decidir qué hacer a continuación.
 - c. También podría tener sentido validar la respuesta de la API para detectar potenciales cambios futuros que pueda romper el contrato entre nuestro código y la API de terceros (schema validation).
2. Una o varias funciones que procesen los datos para calcular el promedio + dispersión a partir de las series temporales de los distintos modelos climáticos.

- a. Hacer unit test, para validar que el código hace lo que crees que hace.
3. Una o varias funciones para realizar el plotting de los datos procesados.

Initial code

Las siguientes variables te son dadas para acelerar el trabajo de desarrollo que tienes que hacer:

```
API_URL = "https://climate-api.open-meteo.com/v1/climate?"

COORDINATES = {
    "Madrid": {"latitude": 40.416775, "longitude": -3.703790},
    "London": {"latitude": 51.507351, "longitude": -0.127758},
    "Rio": {"latitude": -22.906847, "longitude": -43.172896},
}

VARIABLES = "temperature_2m_mean,precipitation_sum,soil_moisture_0_to_10cm_mean"

def main():
    raise NotImplementedError

if __name__ == "__main__":
    main()
```