Universidad Mayor
Carrera Data Science

FIUM1018 - Física Moderna

Profesor: Ariel Norambuena *ariel.norambuena@umayor.cl*

Taller 2

Indicaciones

El Taller de FIUM1018 debe resolverse en horario de clases y de manera grupal. Para el correcto desarrollo, se requiere un computador por grupo, conocimientos de programación en Python y conocer los conceptos vistos en clases (se puede usar todo tipo de material de apoyo: apuntes, diapositivas, libros, páginas web, etc).

Cualquier consulta será respondida por el profesor a modo de guía para inducir la exploración y discusión entre los estudiantes. Cálculos y observaciones adicionales a cada pregunta serán recompensadas con puntos extras al finalizar la actividad.

Lea cuidadosamente las instrucciones previas antes de comenzar el Taller.

Instalación previa

Para poder resolver las preguntas de este taller se recomienda usar la librería Plotly de Python para generar figuras interactivas útiles para analizar datos. Para ello, debes instalar la librería Plotly usando pip:

```
$ pip install plotly==5.7.0
```

o conda

```
conda install -c plotly plotly=5.7.0
```

Luego, para importar la librería Plotly usa los siguientes comandos:

```python
import plotly.io as pio
import plotly.express as px
pio.renderers.default='browser'
```

De esta manera, las figuras serán abiertas en el browser predeterminado del computador. Para cualquier consulta adicional sobre la librería Plotly, se recomienda ir directamente a la página web `https://plotly.com/python/getting-started/#overview`

A continuación ser hará una breve introducción del modelo que usaremos para el Taller 2.

Modelo de radiación de Max Planck y radiación solar

En el año 1900 el Físico alemán Max Planck derivó la teoría de radiación de cuerpo negro usando principios de mecánica estadística y suponiendo que la luz emitida por un objeto a temperatura T estaba cuantizada en paquetes de energía $E_n = nhf$ (n entero, f frecuencia y h constante de Planck). El principal resultado de la teoría de Max Planck es la fórmula para la radiación espectral $B(\lambda, T)$ en términos de la longitud de onda λ y la temperatura T. Para el caso de la radiación emitida por el Sol sobre la Tierra, la fórmula de Max Planck se reduce a la siguiente fórmula [1]:

$$B(\lambda, T) = \left(\frac{R_s}{R_0}\right)^2 \frac{C_1}{\lambda^5} \frac{1}{\exp\left(\frac{C_2}{\lambda T}\right) - 1},$$

donde λ es la longitud de onda de la luz (medida en micrómetros, μm), T es la temperatura del Sol (medida en Kelvin, K), $C_1 = 3{,}7427 \times 10^8$ W $(\mu\text{m})^2$ m^{-2}, $C_2 = 1{,}4399 \times 10^4$ μm K, $R_s = 6{,}9598 \times 10^5$ Km es el radio del disco solar y $R_0 = 149597890$ Km es la distancia media del Sol a la Tierra. La fórmula para $B(\lambda, T)$ permite predecir la potencia espectral emitida del Sol por unidad de área y por unidad de lontidud de onda. De esta manera, la radiación espectral $B(\lambda, T)$ queda en las unidades de W/(m^2 μ m), donde 1W = 1 J/s (Watts) y 1 μm = 10^{-6} m.

Problema

Estás comenzando tu primer proyecto de investigación y el tema de estudio es la radiación solar. Al comienzo del proyecto de investigación tienes acceso a mediciones experimentales realizadas por la NASA sobre la radiación espectral del Sol. Los datos experimentales de la radiación en términos de la longitud de onda están guardados en el documento Excel “SolarRadiation.xlsx”. Para leer los datos, puedes importar la librería Pandas de Python

```
import pandas as pd
```

Para acceder a los datos de radiación espectral $B(\lambda, T)$ y longitud de onda λ, puedes usar el siguiente código:

```
data = pd.read_excel(r'SolarRadiation.xlsx')
lambda_exp = pd.DataFrame(data,columns=
['Longitud de onda (micrometro)'])
B_exp = pd.DataFrame(data,columns= ['Radiacion (W/m^2/micrometro)'])
```

donde “lambda_exp” y “B_exp” son la longitud de onda y radiación experimentales, respectivamente. Para graficar los datos experimentales en el intervalo de (0-5) μm, puedes usar la librería plotly de Python, a través del comando:

```
fig = px.area(data, x="Longitud de onda (micrometro)",
y="Radiacion (W/m^2/micrometro)")
fig.update_xaxes(range=(0, 5),constrain='domain')
fig.show()
```

Una vez ejecutado el código, podrás ir a tu navegador y jugar con la imagen interactiva a través de los botones que aparezcan en la parte superior derecha.

Sigue cuidadosamente las siguientes instrucciones y responde a las preguntas usando programación en Python.

a) Determine la temperatura del Sol usando el modelo de Max Planck. Para ello, defina la radiación espectral $B(\lambda, T)$ como una función de ajuste para los datos experimentales de la NASA, usando la temperatura T como único parámetro de fiteo. Ayuda, primero importa la librería

```
from scipy.optimize import curve_fit
```

Luego, para optimizar el fiteo, puedes usar una temperatura de referencia para mejorar la optimización:

```
popt, _ = curve_fit(objective, lambda_exp, B_exp, Tref)
```

donde "Tref" es alguna temperatura de referencia en Kelvin (recuerda que 300 K $\approx$ 26.85°C) y "objective" es la función que definida por la radiación espectral $B(\lambda, T)$. Prueba con los valores Tref = 500 K, 1000 K y 10000 K y comenta si observas diferencias en el cálculo de la integral. Si el diamante se funde a los $4500°C$, ¿qué pasaría si lanzaramos un trozo de diamante al Sol? Para visualizar los datos y el modelo de manera simultánea usando Plotly, use el comando

```
fig = go.Figure()
fig.add_trace(go.Scatter(x=lambda_exp, y=B_exp, name = 'Experiment'))
fig.add_trace(go.Scatter(x=lambda_fit, y=B_fit,name = 'Model'))
fig.update_xaxes(range=(0, 5),constrain='domain')
fig.show()
```

b) Utilice el modelo de Max Planck para encontrar la longitud de onda donde se encuentra el máximo de la radiación espectral y compárelo con lo que puede estimar al ojo mirando el gráfico de los datos experimentales. Ayuda: debe buscar el índice para el cual se alcanza el máximo de la radiación espectral. Use index = np.where(B_fit==np.amax(B_fit)), donde "B_fit" es el fiteo de la radiación e "index" es la posición donde ocurre el máximo. Indique a qué color corresponde la longitud de onda encontrada.

c) Calcule numéricamente la siguiente integral

$$I = \int_0^\infty B(\lambda, T)\, d\lambda,$$

donde $B(\lambda, T)$ es la radiación espectral hallada en a). Para este cálculo debe definir un nuevo arreglo para la longitud de onda, idealmente que parta en $10^{-2}\mu$m y que termine en un valor muy grande ($\approx 100\mu$m), use cómo mínimo $N = 10^5$ puntos. Se recomienda utilizar el siguiente comando para calcular la integral

```
dlambda = lambda_fit[1]-lambda_fit[0]
I = np.sum(B_fit)*dlambda
```

donde I es la integral dada por la aproximación $I \approx \sum S_{\text{fit}}(\lambda_i)\Delta\lambda$ (suma de Riemann), siendo $\Delta\lambda$ el incremento constante en la longitud de onda. El valor de la integral se conoce como la *constante solar* y cuantifica la cantidad de energía recibida en la Tierra por unidad de tiempo y por unidad de área. Si la constante solar aceptada es $K = 1361$ W/m^2, calcule el error porcentual $|I - K|/K \times 100$ entre el valor estimado (I) y el valor aceptado (K).

Referencias

[1] Muhammad Iqbal, *An introduction to solar radiation*, (Academic Press, Canada, 1983), pag 43.