uniandes

Métodos Computacionales

Tarea 3 - Transoformada de Fourier y Ecuaciones Diferenciales Ordinarias

12-10-2017

uniandes

La solución a este taller debe subirse por SICUA antes de la 11:59PM del Jueves 26 de Octubre del 2017. Los archivos código fuente deben subirse en un único archivo `.zip` con el nombre `NombreApellido_hw3.zip`, por ejemplo yo debería subir el zip `VeronicaArias_hw3.zip`. Este archivo debe descomprimirse en un directorio de nombre `NombreApellido_hw3` que sólo contenga los códigos fuente `fourier1d.py`, `fourier2d.py` y `SIR.py` (5 puntos). Recuerden que es un trabajo TOTALMENTE individual.

1. (10 points) **Transformada de Fourier 1D**
   Los datos del archivo `funcion.dat` son un muestreo a intervalos regulares de una función. Usando su implementación de la transformada de Fourier usted debe determinar la frecuencia de dicha función. Para esto escriban un script en python llamado `fourier1d.py` que debe:

   - Obtener la transformada de fourier de la función (no olvide que debe usar su propia implementación de la transformada).
   - A partir de lo anterior, debe imprimir en consola: `La frecuencia es: FF`, donde FF es el resultado obtenido con el análisis de Fourier.

2. (35 points) **Transformada de Fourier 2D**

   La idea de este ejercicio es que hagan un análisis de cuatro imágenes diferentes utilizando la transformada de Fourier en dos dimensiones. Las imágenes que deben analizar son: `Barcelona.jpg`, `Paris.jpg`, `frac.jpeg`, y `triangulos.png`. Las dos primeras imágenes son planes de Paris y Barcelona y las dos segundas son imágenes fractales.

   Para este ejercicio escriban un script en python llamado `fourier2d.py` que debe:

   - Leer las imágenes y guardarlas en arreglos. Sugerencia: asegúrese de que la imagen esté en escala de grises.
   - Hacer una figura con cuatro subplots, uno por cada imagen, y guardarla sin mostrarla en `imagenes.pdf`.
   - Obtener la transformada de Fourier de las cuatro imágenes (pueden usar los paquetes de scipy o numpy).
   - Hacer una gráfica con cuatro subplots correspondientes a las transformadas de Fourier de las cuatro imágenes, y guardarla sin mostrarla en `transformadas.pdf`.
   - Hacer otra gráfica con cuatro subplots que correspondan a un corte transversal horizontal en el centro de las transformadas de Fourier anteriores, y guardarla sin mostrarla en `cortes_transversales.pdf`.
   - Para la imagen `Barcelona.jpg`, construya un filtro que le permita borrar las vías horizontales del mapa, dejando las verticales. Guarde la imagen obtenida sin vías horizontales en `sin_horizontales.pdf`.

3. (30 points) **Modelo SIR**

El modelo SIR es un modelo matemático que permite describir epidemias (para el caso en que los individuos desarrolan inmunidad y no se tiene en cuenta demografía en el modelo ni otras complicaciones). Una población con N individuos se divide en $S(t)$, que es el número de individuos susceptibles de infectarse, $I(t)$, que es el número de individuos infectados y $R(t)$, que es el número de individuos recuperados que desarrollan inmunidad. La evolución temporal de dichos grupos de individuos se rige por el sistema de ecuaciones diferenciales:

$$\frac{dS}{dt} = -\beta I(t)S(t)$$
$$\frac{dI}{dt} = \beta I(t)S(t) - \gamma I(t)$$
$$\frac{dR}{dt} = \gamma I(t)$$

donde β es la tasa de propagación de la enfermedad, y γ es la tasa de remoción.

La idea de este ejercicio es que resuelvan numéricamente este sistema de ecuaciones para dos casos. Para esto debe escribir un script llamado `SIR.py` que:

- Solucione numéricamente (usando Runge-Kutta o Leap-frog implementado por ustedes) el sistema de ecuaciones descrito anteriormente para N=771, $I(t=0) = 1$ y $S(t=0) = 770$ y para dos casos diferentes: en el caso 1 $\beta = 0.0022$ y $\gamma = 0.45$ y en el caso dos $\beta = 0.001$ y $\gamma = 0.2$.
- Grafique en una sola gráfica con dos subplots las soluciones de los dos casos descritos anteriormente y guárdela sin mostrarla en `SIR.pdf`.
- Encuentre para ambos casos el tiempo (en días) en el cual se llega al máximo de infectados e imprima en consola:
  `* caso 1, tiempo I_max(dias): XX`
  y
  `* caso 2, tiempo I_max(dias): YY`.