Métodos Computacionales

Taller 4

Profesor: Jaime Forero

Fecha de Publicación: *Septiembre 15 de 2015*

Instrucciones de Entrega

La solución a este taller debe subirse por SICUA antes de las 8:30AM del jueves 2 de Octubre del 2015. Si la solución está en SICUA antes de las 8:30AM del domingo 20 de Septiembre 2015 se calificará el taller sobre 130 puntos. Cada punto debe tener la respuesta en un notebook de IPython/Jupyter por separado. Los notebooks deben subirse en un único archivo `.zip` con el nombre `NombreApellido_hw4.zip`, por ejemplo yo debería subir el zip `JaimeForero_hw4.zip`.
Los archivos de datos necesarios se encuentran aquí:
`https://github.com/ComputoCienciasUniandes/MetodosComputacionalesDatos/tree/master/homework/2014-20/hw_5`

1. 30 (40) pt **Limpiando ruido (`limpiando.ipynb`)**

   En clase trabajamos el filtrado de una señal unidimensional quitándole las frecuencias altas y dejando las frecuencias bajas. Ahora vamos a intentar algo similar con una imagen, es decir con una señal bidimensional. Vamos a intentar dos cosas diferentes: en un caso dejar las frecuencias altas y en otro caso dejar las frecuencias bajas.

   Escriba un programa en Python que haga el filtrado de una imagen de dos maneras. La primera que deje pasar las frecuencias bajas; la segunda que deje pasar las frecuencias altas.

   En ambos casos implemente un filtro suave. Ver la siguiente referencia: `http://paulbourke.net/miscellaneous/imagefilter/`.

   Muestre los resultados de aplicar los filtros para las siguientes imágenes

   - `full_moon.jpg`
   - `20_popc_cho009-1.tif`
   - `colesterol-1.tif`
   - `ves_full_150_002-1.tif`

   Cualitativamente hablando: ¿Qué hace cada filtro?.

2. 30 (40) pt **Variabilidad estelar (`variabilidad.ipynb`)**

   Datos de la intensidad de una estrella variable RR-Lyrae se encuentran en `RR_Lyrae_template.dat`.

   a) (10 puntos) Escriba un programa en Python que calcule la misma curva de intensidad cuando se toman en cuenta N componentes de Fourier.

   b) (10 (15) puntos) Prepare gráficas de la curva reconstruída con $N \leq 10$ componentes (i.e. deben aparecer 10 curvas diferentes).

   c) (10 (15) puntos) Prepara una gráfica de χ^2 en función del número de componentes N tomadas en cuenta al momento de hacer la reconstrucción.

3. 40 (50) pt **Círculos (`circulos.ipynb`)**

En el archivo `BAO.dat` se encuentran posiciones en un plano $x - y$. Estos puntos corresponden a la superposición de diferentes círculos más un fondo de puntos distribuidos aleatoriamente a partir de una distribución homogénea.

Encuentre el diámetro de estos círculos a partir de métodos que utilicen la transformada de Fourier.

Ayuda: Función de autocorrelación
`http://mathworld.wolfram.com/Autocorrelation.html`
`http://mathworld.wolfram.com/Wiener-KhinchinTheorem.html`