

La solución a este taller debe subirse por SICUA antes de las 10:00PM del lunes 10 de julio de 2017. Los archivos código fuente deben subirse en un único archivo `.zip` con el nombre `NombreApellido_hw2.zip`, por ejemplo yo debería subir el zip `VeronicaArias_hw2.zip`. Este archivo debe descomprimirse en un directorio de nombre `NombreApellido_hw2` que sólo contenga los códigos fuente `PCA.py` e `fourier.py` (10 puntos). Recuerden que es un trabajo individual.

### 1. (35 points) **PCA**

Principal component analysis (PCA) es un método que permite reducir la dimensionalidad de un problema con muchas variables no independientes y que además permite explorar correlaciones entre dichas variables. La idea de este ejercicio es que usen datos del banco mundial y miren si hay correlaciones entre cinco variables (impuestos, costos de empezar un negocio, desempleo femenino, desempleo masculino y cociente entre mujeres y hombres en la fuerza laboral) para datos de diferentes países. Los datos se encuentran en `DatosBancoMundial5.csv`.

Para este ejercicio deben escribir un script de python (llamado `PCA.py`) con varias funciones que les permitan leer, organizar, hacer un análisis de componente principal y graficar los datos. El script debe:

- Leer los archivos de datos y guardar las variables relevantes en arrays.
- Normalizar los datos
- Graficar los datos y guardar la gráfica anterior (sin mostrarla) en `ExploracionDatos.pdf`
- Calcular la matriz de covarianza para los datos anteriores usando la implementación que hicieron en clase.
- Obtener e imprimir en la consola los DOS componentes principales en orden. Se debe imprimir **el componente principal es: a**, **el segundo componente principal es: b**, donde a y b son los vectores encontrados.
- Graficar los datos nuevamente en el sistema de referencia de los dos componentes principales. Esta gráfica debe ser clara, con ejes debidamente rotulados.
- Guardar la gráfica anterior (sin mostrarla) en `PCAdatos.pdf`
- Hacer una grafica donde se puedan ver las agrupaciones de las variables originales en el sistema de referencia de los dos componentes principales.
- Guardar la gráfica anterior (sin mostrarla) en `PCAvARIABLES.pdf`
- A partir de la gráfica anterior imprima **las variables que están correlacionadas son XX** donde XX corresponde a los grupos de variables encontrados.

### 2. (35 points) **Notas musicales.**

En el archivo `Do.wav` y `Sol.wav` están los sonidos producidos por un piano cuando se tocan las notas Do y Sol respectivamente. La idea de este ejercicio es usar la transformada de Fourier para hacer análisis espectrales de estos archivos y modificarlos.

Para este ejercicio deben escribir una rutina en python (llamada `fourier.py`) con varias funciones que les permitan leer y analizar los datos. Es interesante y recomendable oír los archivos

que se producen en este ejercicio y comparar los sonidos con el sonido original. Recuerden que la transformada de Fourier tiene una parte real y una imaginaria. Para las gráficas de frecuencias sólo interesa la magnitud y las frecuencias positivas, ya que la transformada es simétrica con respecto al cero de las frecuencias. Sin embargo, para hacer los distintos filtros deben tener en cuenta todo el espectro y sus compenetes reales e imaginarias.

La rutina de Python `FourierAnalysis.py` debe:

- Leer los archivos de datos `Do.wav` y `Sol.wav` y extraer la información relevante. Nota: se recomienda usar `wavfile` de `scipy.io`
- Debe usar su propia implementación de la transformada de Fourier.
- Hacer una función que filtre y elimine la frecuencia con mayor amplitud.
- Hacer una función que sea un filtro pasa bajos y que elimine todas las frecuencias mayores a 1000Hz.
- Para los datos de `Do.wav`, hacer una gráfica con tres "sub-plots" (uno para los datos originales y uno para cada filtro) que muestren en el espacio de frecuencias los resultados de los filtros anteriores y los datos originales.
- Guardar la gráfica anterior (sin mostrarla) en `DoFiltros.pdf`
- Cada nota musical tiene una frecuencia característica (la frecuencia fundamental) y sus armónicos (múltiplos enteros de la frecuencia fundamental). En este caso la frecuencia fundamental de la nota Do es 260Hz y la de la nota sol es 391Hz aproximadamente. La idea ahora es que a partir de los datos de la nota Do y cambiando en su script la información de la tasa de sampleo logren que artificialmente la frecuencia fundamental pase de ser 260Hz a 391Hz. Comparen sus resultados con los de la transformada para la nota Sol en una sola gráfica que deben guardar sin mostrarla en `DoSol.pdf`
- A partir de los resultados de los filtros y del cambio artificial de frecuencia, reconstruir para cada caso los datos de la onda correspondiente a cada uno de los espectros de frecuencias y guardarlos en archivos `Do_pico.wav`, `Do_pasabajos.wav` y `DoSol.wav`. Estos archivos deben poderse reproducir con un reproductor de sonido. Mirando las graficas en el espacio de fourier, oigan los diferentes archivos y traten de detectar las diferencias.