

Instrucciones de Entrega

La solución a este taller debe subirse por SICUA antes de las 8:30AM del jueves 11 de Septiembre del 2015. Si la solución está en SICUA antes de las 8:30AM del domingo 6 de Septiembre 2015 se calificará el taller sobre 120 puntos. Los tres archivos código fuente deben subirse en un único archivo `.zip` con el nombre `NombreApellido_hw2.zip`, por ejemplo yo debería subir el zip `JaimeForero_hw2.zip`. Todos los algoritmos deben ser implementados con funciones básicas de python. Librerías externas solamente son permitidas al momento de escribir/leer datos o hacer gráficas.

1. 25 (35) pt **Patrones Ocultos** El siguiente archivo `hands_on/notas_fisicaII_201410.dat` del repositorio `MetodosComputacionalesDatos` contiene datos de notas finales de un curso de Física II. Las tres primeras columnas corresponden a parciales, la siguiente a la nota en complementaria y la quinta al examen final. Haga un análisis de PCA sobre esas cinco columnas de tal manera que se puedan imprimir de manera ordenada los autovalores y autovectores. El análisis solamente debe tener en cuenta a las personas que presentaron todos los parciales.

```
import notas
valores, vectores = notas.pca("notas_fisica_II_201410.dat")
for val, vec in zip(valores, vectores):
    print val, vec
```

2. 25 (35) pt **Ciclo Solar** El siguiente archivo `hands_on/solar/monthrg.dat` tiene datos del número de manchas solares cada mes desde 1610. La primera columna corresponde al año de la medición, la segunda al mes, la tercera al número de días que se pudo observar el número de manchas y la cuarta al número promedio de manchas. Si hay un `-99` en la cuarta columna quiere decir que el número de manchas promedio se desconoce para ese mes.

Asumiendo que este ciclo puede describirse por la siguiente función:

$$f(t) = a \cos(2\pi t + b) + c$$

donde t es el tiempo en años, escriba una función que tome los datos y haga un ajuste por mínimos cuadrados a los parámetros libres de la siguiente manera

```
import numpy as np.
import solar
[a,b,c] = solar.ajuste("monthrg.dat", min=1900, max=1980)
print a, b, c
```

3. 50 (60) pt **Caras en PCA** Como aplicación del método de PCA van a implementar rutinas sencillas para programas de reconocimiento facial.

Vamos a tomar como punto de partida el conjunto de la base de datos de ATT <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>.

Implemente una función que encuentre los autovalores y autovectores correspondientes a las caras en la base de datos y grafica las primeras n (ordenadas en orden decreciente de importancia según los autovalores) en un archivo pdf.

La función debe funcionar de la siguiente manera

```
import caras
[valores, vectores] = caras.pca(pathin="/home/forero/data/at")
n_faces = 10
caras.make_plot(valores, vectores, n=n_faces, fileout="eigenfaces.pdf")
```