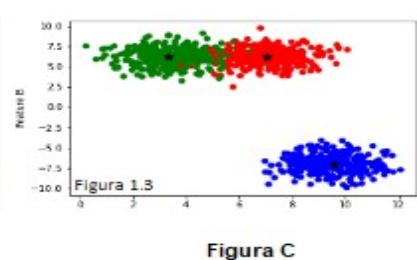
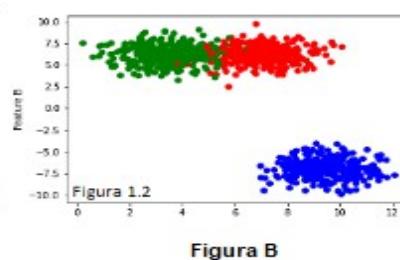
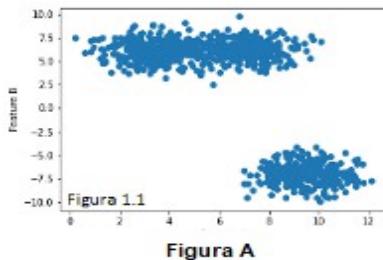


Laboratorio #3 - INFO1157

By Alberto Caro

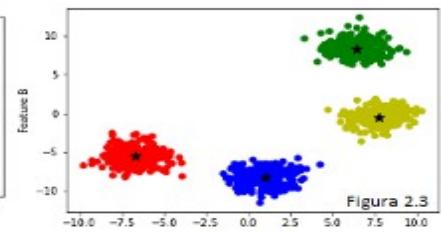
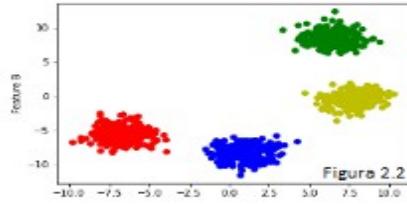
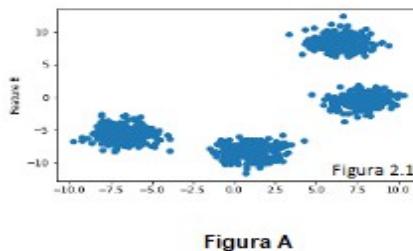
[Parte A]

1.- Aplique KMeans de **sklearn python** y responda las siguientes preguntas. [2 Puntos]



- Dado los datos de entrenamiento de **A.npy** y **_.npy** obtenga el gráfico de la **Figura A**.
- Obtenga el gráfico de la **Figura B** que resalte los **blobs** mediante colores.
- Obtenga los **Cluster** y **Centroides** utilizando **KMeans**. Resalte los centroides con una “Estrella” de color negra como se aprecia en la **Figura C**.
- ¿Cuáles son las etiquetas de la data? ¿Cuántos datos son?
- Haga una predicción del siguiente data test = [[2,5],[3.2,6.5],[7,2.5],[9,3.2],[9,-6],[11,-8]].
- ¿A qué clase pertenecen cada uno de los data test anteriores?

2.- Aplique **MeanShift** de **sklearn python** y responda las siguientes preguntas.[2 Puntos]

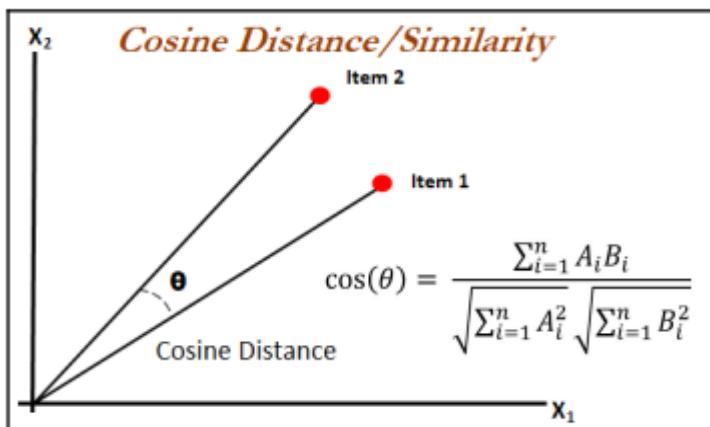


- Dado los datos de entrenamiento de **X.npy** y **_.npy** obtenga el gráfico de la **Figura A**.
- Obtenga el gráfico de la **Figura B** que resalte los **blobs** mediante colores.
- Obtenga los **Cluster** y **Centroides** utilizando **MeanShift**. Resalte los centroides con una “Estrella” de color negra como se aprecia en la **Figura C**.
- ¿Cuáles son las etiquetas de la data? ¿Cuántos datos son?
- Haga una predicción del siguiente data test = [[-7,-6],[1.5,-6.5],[7.9,0.5],[5.5,10]].
- ¿A qué clase pertenecen cada uno de los data test anteriores?

3.- Se tiene la siguiente **Tabla de Pesos y Alturas**. Obtenga una **Regresión Lineal**, mediante **linear_model** de **sklearn**, para predecir el peso dada la altura de una persona. Utilice el siguiente data set para test (Altura) : [1.58m, 1.62m, 1.69m, 1.76m, 1.82m]. Calcule además, el **RSS** del predictor. [1 Punto]

Peso (Kg)	60,0	65,0	72,3	75,0	80,0
Altura (m)	1,60	1,65	1,70	1,73	1,80

4.- Calcule la medida de similaridad utilizando la **Distancia del Coseno**, dado los siguientes vectores:
[2 Puntos]



Vectores

A = [2,1,0,2,0,1,1,1] B = [2,1,1,1,1,0,1,1]
P = [1,2,3,0,4,6,7,9] Q = [2,4,3,1,8,2,4,1]
S = [2,1,4,7,1,4,5,6] T = [3,3,3,6,1,1,7,8]

| INVESTIGUE !

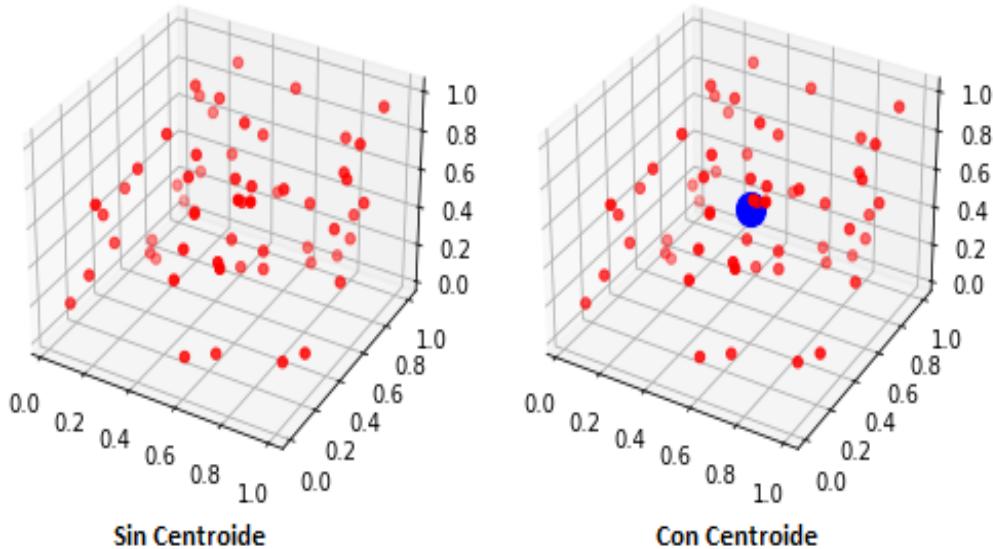
- Explique claramente sus resultados de **Cos(A,B)**, **Cos(P,Q)**, **Cos(S,T)**.
- Cuáles son los ángulos de las distancias anteriores?
- Si $\theta = 0$ rad, ¿qué significa en Distancia del Coseno?
- Si $\theta = \pi/2$ rad, ¿qué significa en Distancia del Coseno?

[Parte B]

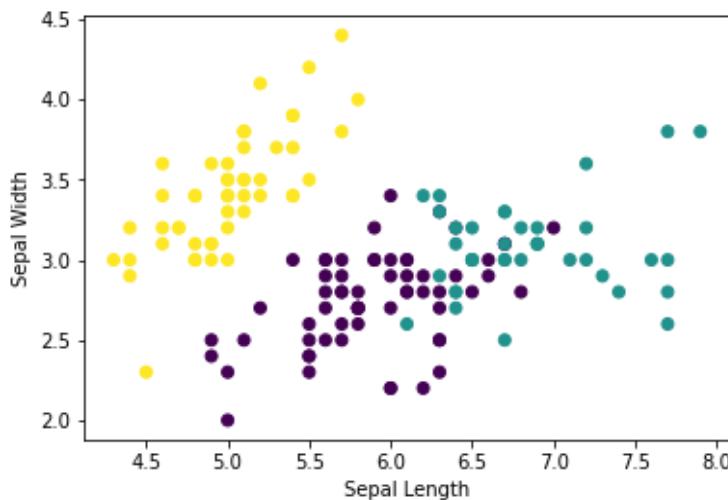
5.- Utilizando **Anaconda**, grafique el **Centroide** de color **Azul**. Utilice el archivo **puntos_3d.npy**. Utilice un **DataFrame** con **3 columnas**. **[1 Punto]**

```
d3 = <- puntos_3d.npy
df = pd.DataFrame( )
df.columns = [....]
```

| Completar !



6.- Utilizando **Anaconda**, complete el script para visualizar el siguiente gráfico. Utilice el **data set** de flores **iris** de Anaconda [2 Puntos]



```
from sklearn.datasets import load_iris  
from sklearn.cluster import KMeans  
iris = load_iris()
```

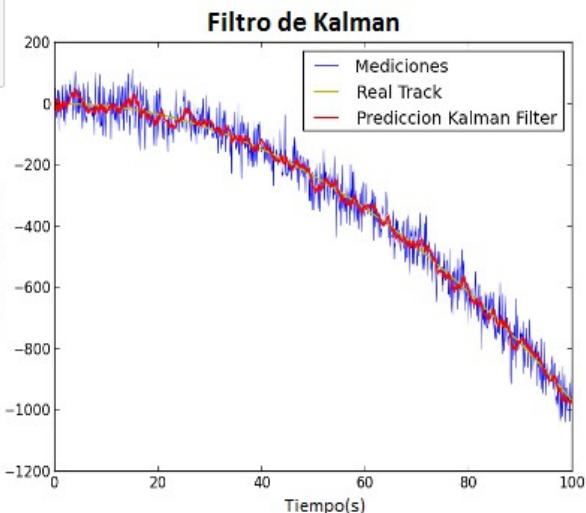
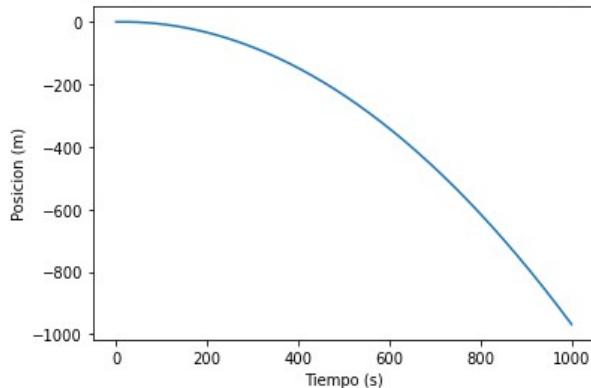
i Completar !

7.- Aplique **Filtro de Kalman (Investigar)** para predecir la posición de una partícula que se mueve siguiendo la siguiente trayectoria: [3 Puntos]

```
In [84]: dt = 0.1  
t = np.arange(0, 100, dt)  
pos = 0.1*(3*t - t**2)
```

```
In [86]: plt.xlabel('Tiempo (s)', fontsize=10)  
plt.ylabel('Posicion (m)', fontsize=10)  
plt.plot(pos)
```

```
Out[86]: <matplotlib.lines.Line2D at 0x2636d35be0>
```



8.- Aplique el **Filtro Complementario** a la predicción del **Filtro de Kalman** para suavizar la curva. Seleccione un **Alpha** que produzca el mejor resultado. Grafique. [1 Punto]

¡Investigue!

Fecha de entrega y defensa Miércoles 19 de Noviembre desde 09:00 - 16:00

<< Trabajo Grupo de 2 personas o individual >>