

# Uso del programa

Lenguajes:

- Python

## ##### ENTRENAMIENTO DE K-MEDIAS #####

En la iteración 0 los centroides son:

```
[[4.5 3. 4. 0.]  
 [6.8 3.4 4.6 0.7]]
```

En la iteración 1 los centroides son:

```
[[5.10395866 3.16786413 2.09623961 0.49170174]  
 [5.95903834 3.00008175 3.83607486 1.15529124]]
```

En la iteración 2 los centroides son:

```
[[5.00995989 3.3798558 1.53281933 0.27036345]  
 [5.96277094 2.8050755 4.26341252 1.32311517]]
```

Hemos terminado en la iteración 3, con los centroides:

```
[[5.00442052 3.40156079 1.48823286 0.25325648]  
 [5.97356112 2.79282529 4.30323088 1.33817883]]
```

## ##### TEST DE K-MEDIAS #####

El archivo TestIris01.txt pertenece al centroide situado en  
[5.00442052 3.40156079 1.48823286 0.25325648] -> Iris-setosa

El archivo TestIris02.txt pertenece al centroide situado en  
[5.97356112 2.79282529 4.30323088 1.33817883] -> Iris-versicolor

El archivo TestIris03.txt pertenece al centroide situado en  
[5.00442052 3.40156079 1.48823286 0.25325648] -> Iris-setosa

## ##### ENTRENAMIENTO DE BAYES #####

Las covariantes obtenidas son:

```
[[[0.121764 0.098292 0.015816 0.010336]  
 [0.098292 0.142276 0.011448 0.011208]  
 [0.015816 0.011448 0.029504 0.005584]  
 [0.010336 0.011208 0.005584 0.011264]]  
  
[[[0.261104 0.08348 0.17924 0.054664]  
 [0.08348 0.0965 0.081 0.04038]  
 [0.17924 0.081 0.2164 0.07164]  
 [0.054664 0.04038 0.07164 0.038324]]]
```

## ##### TEST DE BAYES #####

El archivo TestIris01.txt pertenece a la clase  
Iris-setosa

El archivo TestIris02.txt pertenece a la clase  
Iris-versicolor

El archivo TestIris03.txt pertenece a la clase  
Iris-setosa

#### ##### ENTRENAMIENTO DE LLOYD #####

##### ITERACIÓN 1.

Antiguo:

```
[[4.5 3. 4. 0. ]  
 [6.8 3.4 4.6 0.7]]
```

Nuevo

```
[[5.20117249 2.72259647 2.92150925 0.83850721]  
 [5.94863296 2.85480474 4.34280351 1.32870362]]
```

##### ITERACIÓN 2.

Antiguo:

```
[[5.20117249 2.72259647 2.92150925 0.83850721]  
 [5.94863296 2.85480474 4.34280351 1.32870362]]
```

Nuevo

```
[[5.0738833 2.90692013 2.36074142 0.61373399]  
 [5.88755879 2.81764599 4.2994147 1.32626847]]
```

##### ITERACIÓN 3.

Antiguo:

```
[[5.0738833 2.90692013 2.36074142 0.61373399]  
 [5.88755879 2.81764599 4.2994147 1.32626847]]
```

Nuevo

```
[[4.97159761 3.11168448 1.93818041 0.46558073]  
 [5.86535053 2.8047434 4.26908682 1.31236313]]
```

##### ITERACIÓN 4.

Antiguo:

```
[[4.97159761 3.11168448 1.93818041 0.46558073]  
 [5.86535053 2.8047434 4.26908682 1.31236313]]
```

Nuevo

```
[[4.97211521 3.28833802 1.62381828 0.33719025]  
 [5.80961609 2.77653533 4.20725372 1.292086 ]]
```

#### ##### TEST DE LLOYD #####

El archivo TestIris01.txt pertenece a la clase  
Iris-setosa

El archivo TestIris02.txt pertenece a la clase  
Iris-versicolor

El archivo TestIris03.txt pertenece a la clase  
Iris-setosa

##### ITERACIÓN 5.

Antiguo:

```
[[4.97211521 3.28833802 1.62381828 0.33719025]  
 [5.80961609 2.77653533 4.20725372 1.292086 ]]
```

Nuevo

```
[[4.95790845 3.37684157 1.46928904 0.25177192]  
 [5.74436724 2.75173642 4.09720967 1.27477289]]
```

##### ITERACIÓN 6.

Antiguo:

```
[[4.95790845 3.37684157 1.46928904 0.25177192]  
 [5.74436724 2.75173642 4.09720967 1.27477289]]
```

Nuevo

```
[[4.95783523 3.37729769 1.46849263 0.25133169]  
 [5.74403096 2.75160861 4.09664253 1.27468366]]
```

##### ITERACIÓN 7.

Antiguo:

```
[[4.95783523 3.37729769 1.46849263 0.25133169]  
 [5.74403096 2.75160861 4.09664253 1.27468366]]
```

Nuevo

```
[[4.95783486 3.37730005 1.46848853 0.25132942]  
 [5.74402923 2.75160795 4.09663961 1.2746832 ]]
```

##### ITERACIÓN 8.

Antiguo:

```
[[4.95783486 3.37730005 1.46848853 0.25132942]  
 [5.74402923 2.75160795 4.09663961 1.2746832 ]]
```

Nuevo

```
[[4.95783485 3.37730006 1.46848851 0.25132941]  
 [5.74402922 2.75160795 4.09663959 1.2746832 ]]
```

##### ITERACIÓN 9.

Antiguo:

```
[[4.95783485 3.37730006 1.46848851 0.25132941]  
 [5.74402922 2.75160795 4.09663959 1.2746832 ]]
```

Nuevo

```
[[4.95783485 3.37730006 1.46848851 0.25132941]  
 [5.74402922 2.75160795 4.09663959 1.2746832 ]]
```

Uso:

1. Instalar python e integrarlo en CMD

<https://www.python.org/downloads/>

o, instalar Anaconda y usar Jupyter Notebook (recomendado porque podemos compilar por bloques)

<https://www.anaconda.com/>

2. Instalar librerías

- Pandas: `pip install pandas`
- Numpy: `pip install numpy`
- Matplotlib: `pip install matplotlib`

3. Crear:

- 3.1. Archivo *entrenamiento.txt* dentro de la misma carpeta en la que están los archivos .py (Python) o .ipynb (Anaconda - Jupyter Notebook)

Ejemplo: *entrenamiento.txt*

```
Línea 1:      5.1,3.5,1.4,0.2,Iris-setosa
               ...
Línea 50:     5.0,3.3,1.4,0.2,Iris-setosa
Línea 51:     7.0,3.2,4.7,1.4,Iris-versicolor
               ...
Línea 100:    5.7,2.8,4.1,1.3,Iris-versicolor
```

*NOTA: Si se usa Jupyter Notebook, este archivo se puede omitir, siempre y cuando se cree un np.array con valores, por ejemplo:*

Teoría: Tema 04 - 3, pág 19

```
X = np.array([
    [1, 1],
    [1, 3],
    [1, 5],
    [1, 2],
    [2, 3],
    [2, 3],
    [6, 4],
    [6, 1],
    [7, 3],
    [7, 5],
])

V = np.array([
    [1.0, 4.0],
    [7.0, 2.0],
])
```

- 3.2. Carpeta 'test' con varios .txt (nombre indiferente) que tendrán un único ejemplo en una línea. Esta carpeta debe estar situada junto a los archivos .py (Python) o .ipynb (Anaconda - Jupyter Notebook)

**Ejemplo: TestIris01.txt**

Línea Única: 5.1,3.5,1.4,0.2,Iris-setosa

**Ejemplo: CualquierNombre.txt**

Línea Única: 6.9,3.1,4.9,1.5,Iris-versicolor

*NOTA: Por consistencia, los nombres de los archivos deben tener sentido.*

- 3.3. [OPCIONAL] Carpeta test\_bayes con varios .txt (nombre indiferente) que tendrán un único ejemplo en una línea. Esta carpeta debe estar situada junto a los archivos .py (Python) o .ipynb (Anaconda - Jupyter Notebook)

**Ejemplo: CualquierNombre.txt**

Línea Única: 3,5,'C1'

*NOTA: Por consistencia, los nombres de los archivos deben tener sentido.*

- 3.4. [OPCIONAL] Carpeta test\_kmedias con varios .txt (nombre indiferente) que tendrán un único ejemplo en una línea. Esta carpeta debe estar situada junto a los archivos .py (Python) o .ipynb (Anaconda - Jupyter Notebook)

**Ejemplo: CualquierNombre.txt**

Línea Única: 6.5,3.2,C1

*NOTA: Por consistencia, los nombres de los archivos deben tener sentido.*

- 3.5. [OPCIONAL] Carpeta test\_lloyd con varios .txt (nombre indiferente) que tendrán un único ejemplo en una línea. Esta carpeta debe estar situada junto a los archivos .py (Python) o .ipynb (Anaconda - Jupyter Notebook)

**Ejemplo: CualquierNombre.txt**

Línea Única: 1.3,3.3,C1

*NOTA: Por consistencia, los nombres de los archivos deben tener sentido.*

4. Abrir el terminal y escribir: `python .\main.py` o abrir el archivo `main.ipynb` en Anaconda, el archivo llamará a `Bayes.py`, `Kmedias.py`, `Lloyd.py` y mostrará los resultados de los parámetros

```
##### ENTRENAMIENTO DE K-MEDIAS #####

En la iteración 0 los centroides son:
[[4.5 3. 4. 0. ]
 [6.8 3.4 4.6 0.7]]

En la iteración 1 los centroides son:
[[5.10395866 3.16786413 2.09623961 0.49170174]
 [5.95903834 3.00008175 3.83607486 1.15529124]]

En la iteración 2 los centroides son:
[[5.00995989 3.3798558 1.53281933 0.27036345]
 [5.96277094 2.8050755 4.26341252 1.32311517]]

Hemos terminado en la iteración 3, con los centroides:
[[5.00442052 3.40156079 1.48823286 0.25325648]
 [5.97356112 2.79282529 4.30323088 1.33817883]]

##### TEST DE K-MEDIAS #####
El archivo TestIris01.txt pertenece al centroide situado en
[5.00442052 3.40156079 1.48823286 0.25325648] -> Iris-setosa

El archivo TestIris02.txt pertenece al centroide situado en
[5.97356112 2.79282529 4.30323088 1.33817883] -> Iris-versicolor

El archivo TestIris03.txt pertenece al centroide situado en
[5.00442052 3.40156079 1.48823286 0.25325648] -> Iris-setosa

##### ENTRENAMIENTO DE BAYES #####

Las covariantes obtenidas son:
[[[0.121764 0.098292 0.015816 0.010336]
 [0.098292 0.142276 0.011448 0.011208]
 [0.015816 0.011448 0.029504 0.005584]
 [0.010336 0.011208 0.005584 0.011264]]

 [[0.261104 0.08348 0.17924 0.054664]
 [0.08348 0.0965 0.081 0.04038 ]
 [0.17924 0.081 0.2164 0.07164 ]
 [0.054664 0.04038 0.07164 0.038324]]]

##### TEST DE BAYES #####
El archivo TestIris01.txt pertenece a la clase
Iris-setosa

El archivo TestIris02.txt pertenece a la clase
Iris-versicolor

El archivo TestIris03.txt pertenece a la clase
Iris-setosa
```

```
##### ENTRENAMIENTO DE LLOYD #####

ITERACIÓN 1.
Antiguo:
[[4.5 3. 4. 0. ]
 [6.8 3.4 4.6 0.7]]

Nuevo
[[5.20117249 2.72259647 2.92150925 0.83850721]
 [5.94863296 2.85480474 4.34280351 1.32870362]]

-----

ITERACIÓN 2.
Antiguo:
[[5.20117249 2.72259647 2.92150925 0.83850721]
 [5.94863296 2.85480474 4.34280351 1.32870362]]

Nuevo
[[5.0738833 2.90692013 2.36074142 0.61373399]
 [5.88755879 2.81764599 4.2994147 1.32626847]]

-----

ITERACIÓN 3.
Antiguo:
[[5.0738833 2.90692013 2.36074142 0.61373399]
 [5.88755879 2.81764599 4.2994147 1.32626847]]

Nuevo
[[4.97159761 3.11168448 1.93818041 0.46558073]
 [5.86535053 2.8047434 4.26908682 1.31236313]]

-----

ITERACIÓN 4.
Antiguo:
[[4.97159761 3.11168448 1.93818041 0.46558073]
 [5.86535053 2.8047434 4.26908682 1.31236313]]

Nuevo
[[4.97211521 3.28833802 1.62381828 0.33719025]
 [5.80961609 2.77653533 4.20725372 1.292086 ]]

-----

ITERACIÓN 5.
Antiguo:
[[4.97211521 3.28833802 1.62381828 0.33719025]
 [5.80961609 2.77653533 4.20725372 1.292086 ]]

Nuevo
[[4.95790845 3.37684157 1.46928904 0.25177192]
 [5.74436724 2.75173642 4.09720967 1.27477289]]
```

```
-----
ITERACIÓN 6.
Antiguo:
[[4.95790845 3.37684157 1.46928904 0.25177192]
 [5.74436724 2.75173642 4.09720967 1.27477289]]

Nuevo
[[4.95783523 3.37729769 1.46849263 0.25133169]
 [5.74403096 2.75160861 4.09664253 1.27468366]]

-----
ITERACIÓN 7.
Antiguo:
[[4.95783523 3.37729769 1.46849263 0.25133169]
 [5.74403096 2.75160861 4.09664253 1.27468366]]

Nuevo
[[4.95783486 3.37730005 1.46848853 0.25132942]
 [5.74402923 2.75160795 4.09663961 1.2746832 ]]

-----
ITERACIÓN 8.
Antiguo:
[[4.95783486 3.37730005 1.46848853 0.25132942]
 [5.74402923 2.75160795 4.09663961 1.2746832 ]]

Nuevo
[[4.95783485 3.37730006 1.46848851 0.25132941]
 [5.74402922 2.75160795 4.09663959 1.2746832 ]]

-----
ITERACIÓN 9.
Antiguo:
[[4.95783485 3.37730006 1.46848851 0.25132941]
 [5.74402922 2.75160795 4.09663959 1.2746832 ]]

Nuevo
[[4.95783485 3.37730006 1.46848851 0.25132941]
 [5.74402922 2.75160795 4.09663959 1.2746832 ]]

-----
##### TEST DE LLOYD #####
El archivo TestIris01.txt pertenece a la clase
Iris-setosa

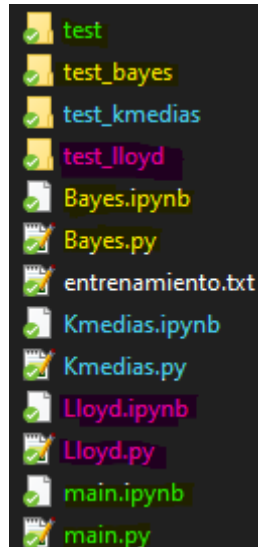
El archivo TestIris02.txt pertenece a la clase
Iris-versicolor

El archivo TestIris03.txt pertenece a la clase
Iris-setosa
```



Notas adicionales:

- La estructura de directorio debe ser como esta:



- Se pueden probar los distintos módulos por separado de una forma sencilla: Abrir el terminal y escribir:
  - `python .\Bayes.py`
  - `python .\Kmedias.py`
  - `python .\Lloyd.py`

Y mostrarán resultados de exámenes o diapositivas del curso

- Podría implementarse desde la lectura de un archivo, pero es contraintuitivo leer datos, parámetros, arrays... Y dividirlo en distintos archivos considero que sería más tedioso que modificarlo desde el propio archivo.



- Examen: Mayo 2019 - Bayes:

```
X = np.array([
    [3, 5],
    [5, 5],
    [4, 5],
    [3, 1],
    [1, 3],
    [2, 2],
    [1, 1],
    [2, 3],
    [0, 2],
])
Y = np.array([
    'C1',
    'C1',
    'C1',
    'C2',
    'C2',
    'C2',
    'C3',
    'C3',
    'C3',
])
vName = ['C1', 'C2',
         'C3']
```

Y las líneas son:

```
medias = bayes(X, Y, vName)
testBayes(medias, vName, 'test_bayes')
```

```
##### ENTRENAMIENTO DE BAYES #####
```

Las covariantes obtenidas son:

```
[[[ 0.66666667  0.
    [ 0.          0.
    ]]]
```

```
[[ 0.66666667 -0.66666667]
 [-0.66666667  0.66666667]]
```

```
[[ 0.66666667  0.33333333]
 [ 0.33333333  0.66666667]]]
```

```
##### TEST DE BAYES #####
```

El archivo prueba\_clase1.txt pertenece a la clase  
C1

El archivo prueba\_clase2.txt pertenece a la clase  
C2

El archivo prueba\_clase3.txt pertenece a la clase  
C3

- ```
X = np.array([
    [1, 1],
    [1, 3],
    [1, 5],
    [1, 2],
    [2, 3],
    [2, 3],
    [6, 4],
    [6, 1],
    [7, 3],
    [7, 5],
])

V = np.array([
    [6.7, 3.43],
    [2.39, 2.94]
])

b = 2

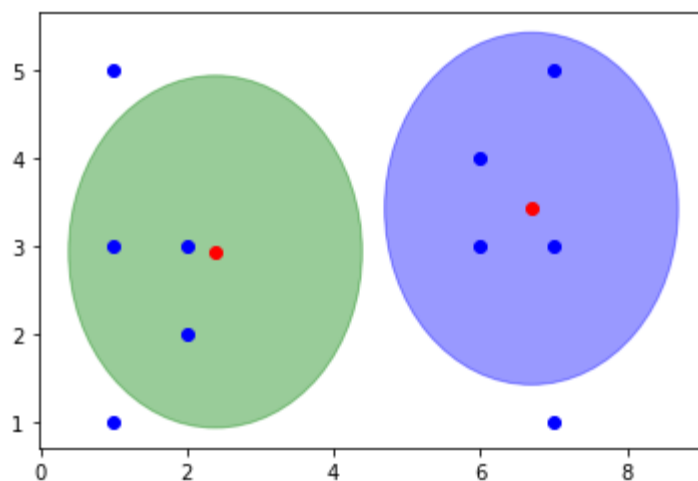
epsilon = .02

vName = ['C1', 'C2']
```

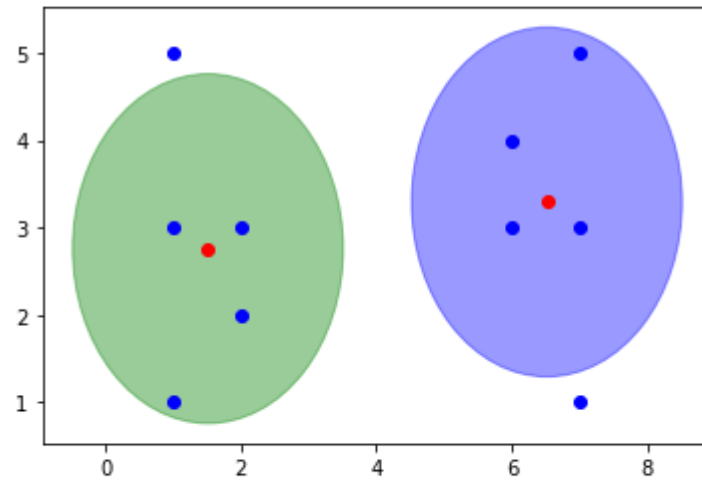
```
vNuevo = kMedias(X, V, b, epsilon)
testKMedias(vNuevo, vName, 'test_kmedias')
```

En la iteración 0 los centroides son:

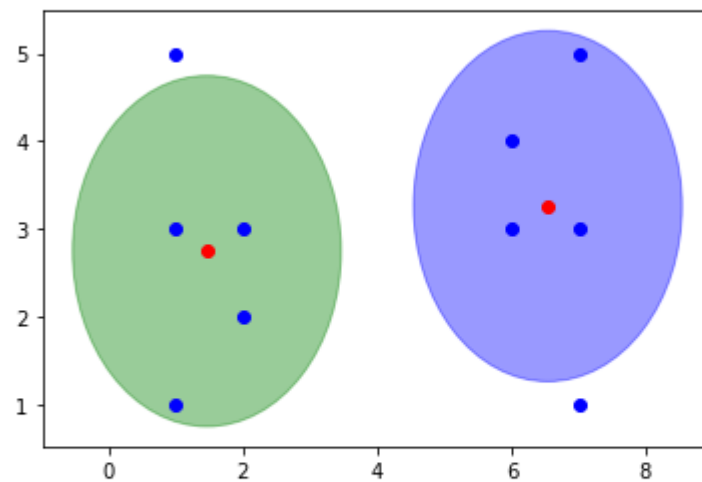
```
[[6.7  3.43]
 [2.39 2.94]]
```



En la iteración 1 los centroides son:  
[[6.52000497 3.29226962]  
[1.50997223 2.75818155]]



Hemos terminado en la iteración 2, con los centroides:  
[[6.54008929 3.25987847]  
[1.46108287 2.7468911 ]]



##### TEST DE K-MEDIAS #####

El archivo prueba\_clase1.txt pertenece al centroide situado en  
[6.54008929 3.25987847] -> C1

El archivo prueba\_clase2.txt pertenece al centroide situado en  
[1.46108287 2.7468911 ] -> C2

*NOTA: Las gráficas solo funcionan con vectores de dos dimensiones*

- ```
X = np.array([
    [1, 1],
    [1, 3],
    [1, 5],
    [1, 2],
    [2, 3],
    [2, 3],
    [6, 4],
    [6, 1],
    [7, 3],
    [7, 5],
])

V = np.array([
    [1.0,
     4.0],
    [7.0,
     2.0],
])

gamma = .1
kMax = 10
epsilon = .1
vName = ['C1',
          'C2']
```

```
vNuevo = lloyd(X, V, gamma, kMax, epsilon)
testLloyd(vNuevo, vName, 'test_lloyd')
```

El archivo `prueba_clase2.txt` pertenece a la clase C2