

### Análisis de Discriminante y Vecinos más cercanos

Iris es un género de plantas rizomatosas de la familia Iridaceae. Dentro de las plantas de Iris existen muchas especies distintas, entre ellas *setosa*, *versicolor* y *virginica*. Se desea desarrollar un clasificador de dichas plantas a partir del tamaño de su sépalo.

#### (a) Exploración de datos:

- Cargar la base de datos utilizando `datasets.load_iris` (sklearn). Utilizar solamente las dimensiones del sépalo. Graficar las muestras en un `scatter`, representando las diferentes especies de Iris con colores distintos.
- Utilice el comando `train_test_split` (sklearn) para definir los conjuntos de entrenamiento y testeo con las proporciones 70 % y 30 %.

#### (b) Clasificación LDA:

- Implementar un algoritmo de LDA a partir de las dimensiones del sépalo para clasificar las plantas de Iris. El código debe estar estructurado de la siguiente manera:


```
class LDA:

    # Inicializar atributos y declarar hiperparámetros
    def __init__(self,...


    # Etapa de entrenamiento
    def fit(self,X,y):


    # Etapa de testeo soft
    def predict_proba(self,X):

    # Etapa de testeo hard
    def predict(self,X):
```

- Superponer al `scatter` las medias y las covarianzas de cada gaussiana (una curva de nivel) del modelo de LDA.
- Superponer al `scatter` las fronteras de decisión. : Funciones como `meshgrid` (numpy) y `contour` (matplotlib) pueden ser útiles para graficar las fronteras.
- Reportar el accuracy de testeo.

(c) *Síntesis*: Utilizando el modelo LDA, generar 30 muestras sintéticas y graficarlas junto a las fronteras. Representar las diferentes especies de Iris con colores distintos.

: Tenga en cuenta que, además de las medias y varianzas, deberá utilizar las probabilidades  $c_k$  aprendidas durante el entrenamiento. Funciones como `random.choice` y `random.multivariate_normal` (numpy) pueden ser útiles.

(d) *Discriminante cuadrático*: Repetir los incisos (b) y (c) utilizando un algoritmo QDA. : Dada la similitud entre ambos algoritmos, se recomienda modificar el código anterior utilizando un hiperparámetro que indique si estamos en un análisis lineal o cuadrático.

(e) *K vecinos más cercanos*: Repetir el inciso (b) utilizando un algoritmo KNN con  $K = 6$  (excepto por las curvas de nivel asociadas a las medias y covarianzas que aquí no corresponden).