

Practica 0

Aprendizaje Automático

Ejercicio 1

-Leer la base de datos de iris que hay en scikit-learn.

Para esto cargamos iris con el comando `ds.load_iris()` la base de datos iris.

-Obtener las características (datos de entrada X) y la clase (y).

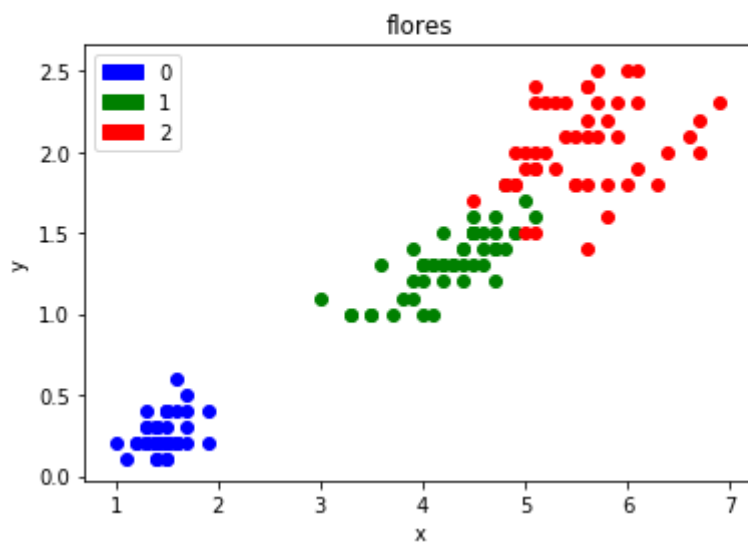
Para almacenar en x las características de las características nos quedaremos con `iris.data` y para la clase que la almacenaremos en `y iris.target`,

-Quedarse con las dos últimas características (2 últimas columnas de X).

Para esto utilizamos las propiedades de las listas de python y con su operador nos quedaremos con las dos ultimas columnas. `[:, 2:]`

-Visualizar con un Scatter Plot los datos, coloreando cada clase con un color diferente (con rojo, verde y azul), e indicando con una leyenda la clase a la que corresponde cada color.

```
In [4]: runfile('/home/jose/Documentos/git/AA/practica0/practica0.py', wdir  
git/AA/practica0')
```



Ejercicio 2

Separar en training (80 % de los datos) y test (20 %) aleatoriamente conservando la proporción de elementos en cada clase tanto en training como en test. Con esto se pretende evitar que haya clases infra-representadas en entrenamiento o test. Nota: el comando `zip` puede ser de utilidad en este apartado

He separado las flores en tres vectores uno por cada clase. Para generar el vector de test tengo que tener en cuenta que al tener el mismo número de muestras de cada clase en la población, para estar igualmente representadas en test tendré que coger el mismo número de ambas. En este caso con se pide un 20% de la población total y esta es de 150 tendré que seleccionar de cada tipo 10 elementos. Genero una lista de números aleatorios de 0 a 49 que son el número de flores de cada tipo que tengo. Tengo que realizar eso para cada tipo de flor. Una vez generadas para test me quedo con las flores dadas por esos índices.

En training meteré todas las flores que no estén en test y con esto tendré el 80% bien representado.

Ejercicio 3

Obtener 100 valores equiespaciados entre 0 y 2π

Se realiza con el comando `np.linspace(0,2*math.pi,100)`

Obtener el valor de $\sin(x)$, $\cos(x)$ y $\sin(x) + \cos(x)$ para los 100 valores anteriormente calculados.

Simplemente utilizamos la función de `math.sin()` y `math.cos()`.

Visualizar las tres curvas simultáneamente en el mismo plot (con líneas discontinuas en negro, azul y rojo).

