Bloque 2
Aprendizaje Automático

Práctica 2:

Sesión2
Aplicación del algoritmo del Perceptron a tareas de clasificación

DOCENCIA VIRTUAL

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Responsable del Tratamiento: Universitat Politècnia de València (UPV)
Finalidad: Prestación del servicio público de educación superior en base al interés público de la UPV (Art. 6.1.e del RGPD).
Ejercicio de derechos y segunda capa informativa: Podrán ejercer los derechos reconocidos en el RGPD y la LOPDGDD de acceso, rectificación, oposición, supresión, etc., escribiendo al correo dpd@upv.es.
Para obtener más información sobre el tratamiento de sus datos puede visitar el siguiente enlace: https://www.upv.es/contenidos/DPD.
Propiedad Intelectual: Uso exclusivo en el entorno del aula virtual.
Queda prohibida la difusión, distribución o divulgación de la grabación de las clases y particularmente su compartición en redes sociales o servicios dedicados a compartir apuntes.
La infracción de esta prohibición puede generar responsabilidad disciplinaria, administrativa y/o civil.

Sesiones de la práctica 2

Sesión 1:

- Familiarizarse con el entorno de trabajo (Google Colab)
- Analizar conjuntos de datos (datasets): **iris**, **digits**, **olivetti**, **openml**

Sesión 2:

- Aplicación del algoritmo del Perceptron a tareas de clasificación: dataset **iris**.
- **Ejercicio: A**plicar Perceptrón a **digits** y **olivetti**

Sesión 3:

- Aplicación de Regresión Logística a tareas de clasificación: dataset **iris**.

Ejemplo de examen:

- Aplicación de Perceptrón y Regresión Logística a un dataset de OpenML.

Sesión 4 (examen):

- Se pedirá la aplicación de Regresión Logística para una tarea diferente de OpenML
- Hay que subir también la solución del Ejercicio

Sesión 2: Aplicación del algoritmo del Perceptron

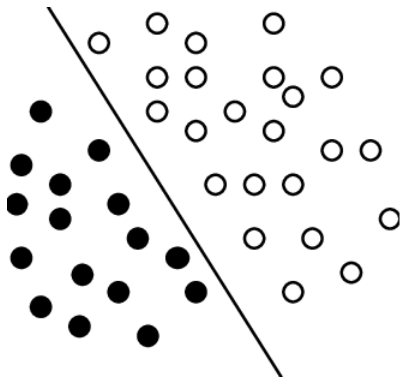
- Un clasificador lineal toma una decisión de clasificación basándose en una **combinación lineal de las características de entrada**.

- Una **función discriminante lineal** se utiliza para dividir el espacio de características en regiones de decisión, clasificando los puntos de acuerdo con sus valores.

$$g_c(x) = w_c^T x + w_{c0}$$

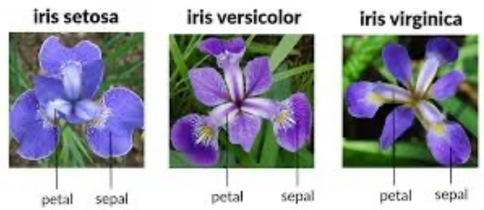
$$g_c(x) = w_{c1}x_1 + w_{c2}x_2 + \cdots + w_{cn}x_n + w_{c0}$$

Donde x son los valores de una muestra, w son los pesos incluido el término independiente.



Sesión 2: Aplicación del algoritmo del Perceptron

Ejemplo en el dataset Iris



- Tenemos la muestra:

x_1: Long. sépalo	x_2: Ancho sépalo	x_3: Long. pétalo	x_4: Ancho pétalo
5.1	3.5	1.4	0.2

- La función lineal tiene la forma:

$$g_c(x) = w_{c0} + w_{c1}x_1 + w_{c2}x_2 + \cdots + w_{cn}x_n$$

- Suponiendo los siguientes pesos:

w_{c0}	w_{c1}	w_{c2}	w_{c3}	w_{c4}
-2.0	0.6	-0.3	1.2	0.5

- El cálculo es: $g_c(x) = (-2.0)(1) + (0.6)(5.1) + (-0.3)(3.5) + (1.2)(1.4) + (0.5)(0.2)$

$$g_c(x) = -2.0 + 3.06 - 1.05 + 1.68 + 0.1 = 1.79$$

Sesión 2: Aplicación del algoritmo del Perceptron

Clasificador lineal

Dado un conjunto de funciones lineales $g_c(x)$ para cada clase $c \in \{1, 2, \ldots, C\}$, donde:

$$\boldsymbol{g_c(x) = w_c^T x + w_{c0}}$$

El clasificador asigna una entrada x a la clase c tal que:

$$\hat{c} = \arg\max_c g_c(x)$$

El clasificador selecciona la clase cuya función discriminante $g_c(x)$ tiene el valor más alto.

Sesión 2: Aplicación del algoritmo del Perceptron

- Ejemplo Iris

Característica	Iris-setosa	Iris-versicolor	Iris-virginica
Término Independiente	-1.5	0.5	0.8
x_1 (Sepal Length)	0.5	-0.3	0.2
x_2 (Sepal Width)	-0.2	0.7	0.4
x_3 (Petal Length)	0.8	0.6	-0.5
x_4 (Petal Width)	0.3	0.2	0.9

$$W = \begin{array}{c} \begin{matrix} g_0(x) & \quad g_1(x) & \quad g_2(x) \end{matrix} \\ \begin{bmatrix} -1.5 & 0.5 & 0.8 \\ 0.5 & -0.3 & 0.2 \\ -0.2 & 0.7 & 0.4 \\ 0.8 & 0.6 & -0.5 \\ 0.3 & 0.2 & 0.9 \end{bmatrix} \end{array}$$

Un clasificador al final es una matriz de pesos

Sesión 2: Aplicación del algoritmo del Perceptron

¿Cómo obtener la matriz? → ¿Como aprendemos los pesos? → Algoritmo Perceptron

Entrada: $\{(\mathbf{x}_n, c_n)\}_{n=1}^{N}, \quad \{\mathbf{w}_c\}_{c=1}^{C}, \quad \alpha \in \mathbb{R}^{>0}$ y $b \in \mathbb{R}$

Salida: $\{\mathbf{w}_c\}^* = \underset{\{\mathbf{w}_c\}}{\arg\min} \sum_n \left[\max_{c \neq c_n} \mathbf{w}_c^t \mathbf{x}_n + b > \mathbf{w}_{c_n}^t \mathbf{x}_n \right]$

Método:

$$[P] = \begin{cases} 1 & \text{si} \quad P = \text{verdadero} \\ 0 & \text{si} \quad P = \text{falso} \end{cases}$$

repetir

para todo dato $\mathbf{x}_n$

$err = \texttt{falso}$

para toda clase c distinta de c_n

si $\mathbf{w}_c^t \mathbf{x}_n + b > \mathbf{w}_{c_n}^t \mathbf{x}_n$: $\mathbf{w}_c = \mathbf{w}_c - \alpha \cdot \mathbf{x}_n$; $err = \texttt{verdadero}$

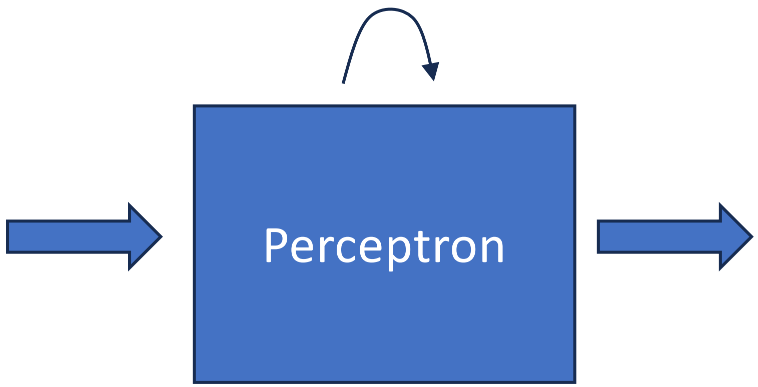
si err: $\mathbf{w}_{c_n} = \mathbf{w}_{c_n} + \alpha \cdot \mathbf{x}_n$

hasta que no queden muestras mal clasificadas
(o se llegue a un máximo de iteraciones prefijado)

Sesión 2: Aplicación del algoritmo del Perceptron

Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
1	5.1	3.5	1.4	0.2	Iris-setosa
2	4.9	3	1.4	0.2	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa
4	4.6	3.1	1.5	0.2	Iris-setosa
5	5	3.6	1.4	0.2	Iris-setosa
6	5.4	3.9	1.7	0.4	Iris-setosa
7	4.6	3.4	1.4	0.3	Iris-setosa
8	5	3.4	1.5	0.2	Iris-setosa
9	4.4	2.9	1.4	0.2	Iris-setosa
10	4.9	3.1	1.5	0.1	Iris-setosa
11	5.4	3.7	1.5	0.2	Iris-setosa
12	4.8	3.4	1.6	0.2	Iris-setosa
13	4.8	3	1.4	0.1	Iris-setosa
14	4.3	3	1.1	0.1	Iris-setosa
15	5.8	4	1.2	0.2	Iris-setosa
16	5.7	4.4	1.5	0.4	Iris-setosa
17	5.4	3.9	1.3	0.4	Iris-setosa
18	5.1	3.5	1.4	0.3	Iris-setosa
19	5.7	3.8	1.7	0.3	Iris-setosa

Muestras etiquetadas



$$W = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\begin{array}{cccc} & g_0(x) & g_1(x) & g_2(x) \\ W = & \multicolumn{3}{c}{\begin{bmatrix} -1.5 & 0.5 & 0.8 \\ 0.5 & -0.3 & 0.2 \\ -0.2 & 0.7 & 0.4 \\ 0.8 & 0.6 & -0.5 \\ 0.3 & 0.2 & 0.9 \end{bmatrix}} \end{array}$$

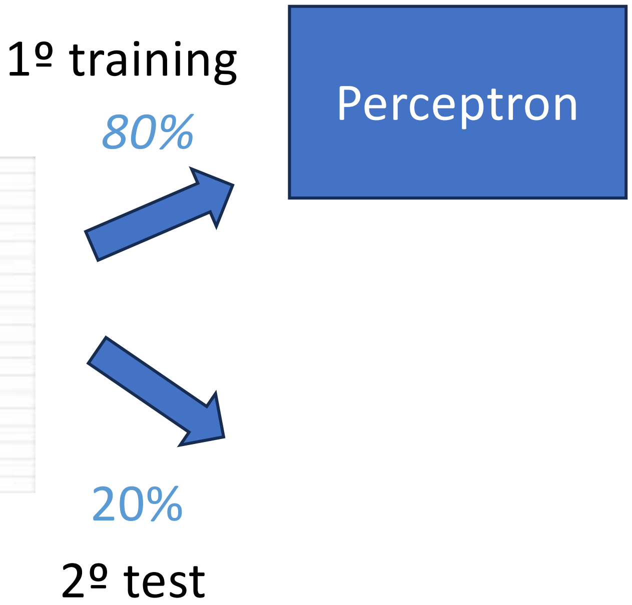
W: Matriz de pesos
K: número iteraciones
E: muestras mal clasificadas

Sesión 2: Aplicación del algoritmo del Perceptron

- **Cálculo del error:** dividir el dataset en dos subconjuntos, uno para **entrenamiento** (**training**) y otro para **pruebas** (**test**)

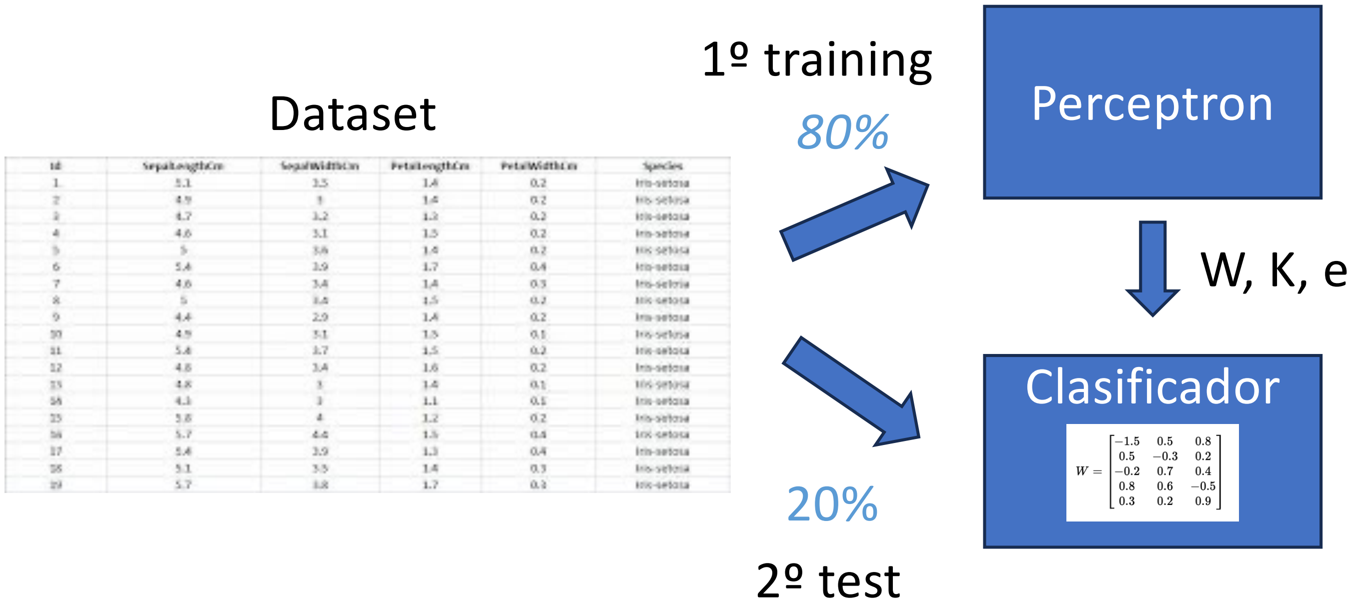
Dataset

Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
1	5.1	3.5	1.4	0.2	Iris-setosa
2	4.9	3	1.4	0.2	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa
4	4.6	3.1	1.5	0.2	Iris-setosa
5	5	3.6	1.4	0.2	Iris-setosa
6	5.4	3.9	1.7	0.4	Iris-setosa
7	4.6	3.4	1.4	0.3	Iris-setosa
8	5	3.4	1.5	0.2	Iris-setosa
9	4.4	2.9	1.4	0.2	Iris-setosa
10	4.9	3.1	1.5	0.1	Iris-setosa
11	5.4	3.7	1.5	0.2	Iris-setosa
12	4.8	3.4	1.6	0.2	Iris-setosa
13	4.8	3	1.4	0.1	Iris-setosa
14	4.3	3	1.1	0.1	Iris-setosa
15	5.8	4	1.2	0.2	Iris-setosa
16	5.7	4.4	1.5	0.4	Iris-setosa
17	5.4	3.9	1.3	0.4	Iris-setosa
18	5.1	3.5	1.4	0.3	Iris-setosa
19	5.7	3.8	1.7	0.3	Iris-setosa



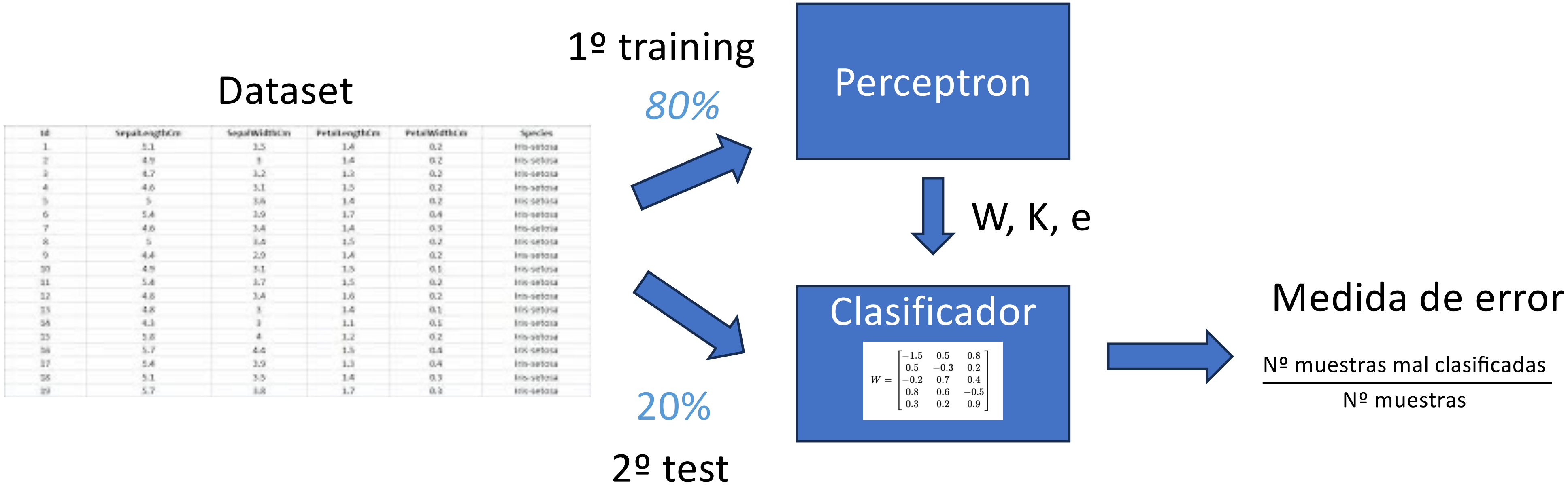
Sesión 2: Aplicación del algoritmo del Perceptron

- **Cálculo del error:** dividir el dataset en dos subconjuntos, uno para **entrenamiento** (**training**) y otro para **pruebas** (**test**)



Sesión 2: Aplicación del algoritmo del Perceptron

- **Cálculo del error:** dividir el dataset en dos subconjuntos, uno para **entrenamiento** (**training**) y otro para **pruebas** (**test**)



Sesión 2: Aplicación del algoritmo del Perceptron

Ajuste del modelo

- **α: tasa de aprendizaje** (Learning Rate): Es un escalar que determina cuánto se ajustan los pesos durante el proceso de aprendizaje.
  - Si α es demasiado grande, el algoritmo puede oscilar o divergir. Si es demasiado pequeño, el aprendizaje será lento.
  - Se puede probar con diferentes α: (.01, .1, 10, 100)
- **b: margen**
  - Es un parámetro adicional que permite ajustar las fronteras de decisión
  - Se puede probar con diferentes b: (.0, .01, .1, 10, 100)

Sesión 2: Aplicación del algoritmo del Perceptron

Probar iris.ipynb

Ejercicio: Aplicar Perceptrón a digits y Olivetti. (entregar el dia del examen)