

Preparación de Datos. Evaluación

- Los datos y sus transformaciones
- Evaluación de una red de neuronas

Los datos

- Variables, datos o patrones de entrada y de salida deseada para aquellas redes que utilicen aprendizaje supervisado.
- Variables, datos o patrones de entrada para aquellas redes que utilicen aprendizaje no supervisado (no hay salida).
- Las redes de neuronas sólo trabajan con atributos numéricos.
- Los atributos no numéricos (nominales) deben discretizarse a valores numéricos. Esta discretización puede influir en los resultados de la red.

Los datos

Aspecto del conjunto de datos

Entradas							Salida Deseada
	X1	X2	X3	X4	X5	X5	Salida
Patrón1	99.969,8	1,857	280,295	0,056	0,012	0,629	402,71
Patrón2	100.332,3	- 1,810	278,534	- 0,104	- 0,080	- 2,991	562,50
Patrón3	99.663,5	- 0,177	275,086	0,006	0,155	5,360	194,62
Patrón4	99.918,8	1,226	278,377	0,062	0,110	4,177	155,94
Patrón5	99.896,1	- 0,177	276,289	- 0,002	0,050	2,886	641,34
Patrón6	99.329,3	2,582	278,464	0,290	0,562	8,042	413,33
Patrón7	99.388,9	1,306	280,543	0,110	0,305	6,789	327,31

PatrónN	98.747,3	- 5,479	280,101	- 0,693	- 0,463	- 5,798	1.968,71

Los datos y sus transformaciones

- Transformaciones de los datos
 - **Normalización de los datos.** Es aconsejable trabajar con datos normalizados en un cierto intervalo, generalmente el intervalo [0,1]. Si bien no es obligatorio, siempre es recomendable pues evita problemas durante el aprendizaje, como la saturación de las neuronas (Perceptron Multicapa).
 - Es aconsejable realizar la normalización para cada variable de manera independiente

$$\text{VarNor}_i = (\text{Var}_i - \text{VMin}_i) / (\text{VMax}_i - \text{VMin}_i)$$

- **Aleatorización de los datos.** Para evitar sesgos en el aprendizaje, siempre es conveniente aleatorizar los patrones de entrenamiento.

Los datos y sus transformaciones

- Otras posibles transformaciones:

Eliminación de atributos irrelevantes, pues no aportan información al problema

Reducción dimensionalidad: Para algunos problemas, los datos de entrada poseen alta dimensión. Es conveniente aplicar técnicas de reducción de dimensionalidad, que bien seleccionan un subconjunto de atributos, bien transforman los datos de entrada en otro conjunto de menor dimensión. No se entra en detalle en los diferentes métodos de reducción de dimensionalidad, pues no es objetivo del curso.

Evaluación de una red de Neuronas

- **Separación del conjunto de entrenamiento y test.**
 - El conjunto de datos disponible se separa en dos subconjuntos: Conjunto de entrenamiento (aprendizaje) y conjunto de test (generalización).
 - Esta división debe realizarse aleatoriamente, o al menos utilizando un mecanismo que garantice que los datos de test están representados en el conjunto de entrenamiento
- **Separación del conjunto de entrenamiento, validación y test. (Practica I)**
 - El conjunto de datos disponible se separa en tres subconjuntos: Conjunto de entrenamiento (aprendizaje), conjunto de validación (parada del aprendizaje y determinación de los hiperparámetros de la red) y conjunto de test (generalización).
 - Esta división debe realizarse aleatoriamente
 - El conjunto de validación puede ser útil para elegir los mejores hiperparámetros (neuronas, capas, número de ciclos, razón de aprendizaje, etc). Si la elección se hace utilizando el conjunto de entrenamiento, podría ocurrir el “sobreaprendizaje”.

Evaluación de una red de Neuronas

- **Validación cruzada.(Práctica II)**

- Para evitar sesgos en los datos de entrenamiento y test. Es muy útil cuando hay pocos datos disponibles
- Se divide el conjunto de datos en k partes o subconjuntos. Supongamos $k=3$ (generalmente $k=10$) ;los subconjuntos A, B, y C. Se realizan k (3) iteraciones:
 - Aprender con A, B y test con C ($T1$ = medida de evaluación con C)
 - Aprender con A, C y test con B ($T2$ = medida de evaluación con B)
 - Aprender con B, C y test con A ($T3$ = medida de evaluación con A)
 - Medida de evaluación final $T = (T1+T2+T3)/3$
- La elección de los hiperparámetros se puede hacer para cada grupo, extrayendo de cada partición de entrenamiento un subconjunto de validación
- Si el número de patrones es escaso, se podría hacer la selección de los hiperparámetros con la media en test para todos los grupos
- Cuando se utiliza VC, el modelo final se construye con todos los datos disponibles

Evaluación de una red de Neuronas

- **Validación cruzada estratificada.**
 - En problemas de clasificación con desbalanceo entre clases, si se utiliza VC es recomendable utilizar la VC estratificada.
 - El procedimiento es igual, pero al dividir los datos en k partes, se deberá mantener la proporción de instancias de las clases del conjunto total en cada una de las partes A, B, y C (en el caso $k=3$)

Por ejemplo, si se dispone de 100 patrones con 2 clases (C1 y C2) y si el número de patrones para la C1 es 75 y para la clase C2 es 25, entonces cuando se dividan los datos en las partes A, B y C, cada parte debe contener 75% de C1 y 25% de C2.
 - Un posible mecanismo: dividir los patrones de cada clase en k partes y posteriormente juntarlos
 - Muchas herramientas y/o librerías de Aprendizaje Automático contienen la VC y VC Estratificada

Evaluación de una red de Neuronas

- **Medida de Evaluación:** dependen de la tarea o problema a resolver.
 - En **problemas de regresión** (aproximación) o predicción la manera más habitual de evaluar la red es mediante el error medio o error cuadrático medio cometido por la red sobre las datos de entrenamiento y test (son equivalentes) :

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - d_i|$$

MAE: Mean Absolute Error

y_i : salida de la red, d_i : salida deseada

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - d_i)^2$$

MSE: Mean Square Error

- En **problemas de clasificación** lo más habitual es evaluar la calidad de la red en base a su precisión predictiva (% de aciertos): el número de patrones del conjunto de entrenamiento o test clasificadas correctamente dividido por el número total de instancias en dicho conjunto
- En problemas de agrupación o clustering se suele medir la cohesión de cada grupo o cluster y la separación entre los grupos (distancia Euclídea)

Evaluación de una red de Neuronas

- A lo hora de decidir si una red resuelve con éxito o no un problema
 - **En regresión:**
 - Un error medio (o error cuadrático medio) aceptable
 - Visualización gráfica de las salidas deseadas y las salidas proporcionadas por la red
 - **En clasificación**
 - Un alto % de aciertos, aunque hay que tener presente:
 - En problemas con M clases, el porcentaje de aciertos debe superar el $100 \cdot 1/M$. De otra manera, sería mejor tirar una moneda (azar) que utilizar el clasificador
 - Si se dispone de una clase con muchos más datos que otra, el porcentaje de aciertos a superar es el porcentaje de datos de la clase mayoritaria. Por ejemplo, si tenemos dos clases (+ y -) y hay 90 datos + y 10 -; un clasificador que prediga siempre + ya acertará en un 90%. Hay que hacerlo mejor que eso.
 - En ocasiones el coste de fallar en una clase no es el mismo que fallar en otra. Por ejemplo, para un clasificador de cáncer si/no, es preferible predecir que una persona tiene cáncer (sin tenerlo) que predecir que no lo tiene (teniéndolo). Para estas situaciones se suele utilizar la matriz de confusión, que contiene falsos positivos y falsos negativos.

Evaluación de una red de Neuronas

Matriz de confusión (problemas de clasificación)

	Clasificado como +	Clasificado como -
Dato realmente +	TP (true positive)	FN (false negative)
Dato realmente -	FP (false positive)	TN (true negative)

- Los datos correctamente clasificados están en la diagonal, los incorrectos fuera de ella.
- El porcentaje de aciertos total es
$$(TP+TN)/(TP+TN+FN+FP)$$
- El porcentaje de aciertos de + es:
$$TP/(TP+FN)$$
- El porcentaje de aciertos – es:
$$TN/(FP+TN)$$