

El perceptrón de capa simple.

Mario A. Ibarra-Manzano

Departamento de Comunicaciones y Electrónica
Facultad de Ingeniería, Mecánica, Eléctrica y Electrónica

Marzo del 2006.
20 de marzo de 2006



Contenido

- 1 Introducción.
- 2 Modelo del perceptron.
- 3 Algoritmo de aprendizaje.
- 4 Aplicaciones del perceptrón.



Contenido

- 1 Introducción.
- 2 Modelo del perceptron.
- 3 Algoritmo de aprendizaje.
- 4 Aplicaciones del perceptrón.



Contenido

- 1 Introducción.
- 2 Modelo del perceptron.
- 3 Algoritmo de aprendizaje.
- 4 Aplicaciones del perceptrón.



Contenido

- 1 Introducción.
- 2 Modelo del perceptron.
- 3 Algoritmo de aprendizaje.
- 4 Aplicaciones del perceptrón.



Introducción

La RNA más utilizada es el perceptrón de múltiples capas (MLP por sus siglas en inglés), este modelo es una variante del perceptrón propuesto por Rosenblatt en 1950. El perceptrón ha generado mucho interés debido a su habilidad para aprender y reconocer patrones simples.



Modelo general.

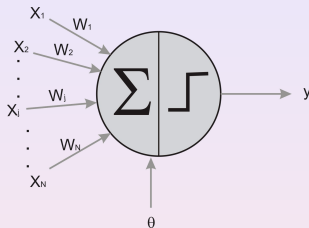


Figura: Modelo un perceptron de capa simple.

$$u(x) = \sum_{i=1}^N w_i x_i + \theta \quad (1)$$

$$y(x) = \begin{cases} 1 & u(x) \geq 0 \\ 0 & u(x) < 0 \end{cases}$$

Modelo de dos entradas.

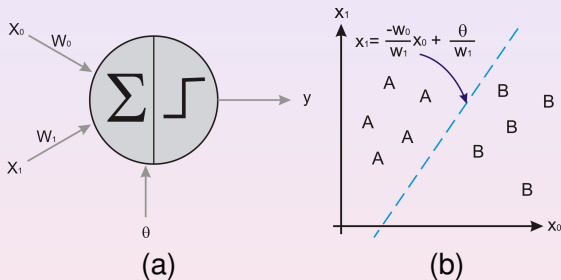


Figura: (a) Modelo de un perceptrón (b) Clasificación mediante un hiperplano

Demostración.

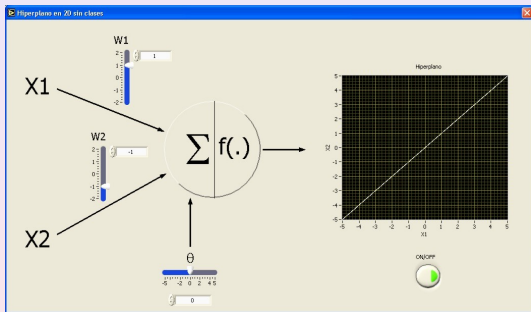


Figura: Modelo del perceptrón de 2 entradas.

Algoritmo de entrenamiento del perceptrón simple.

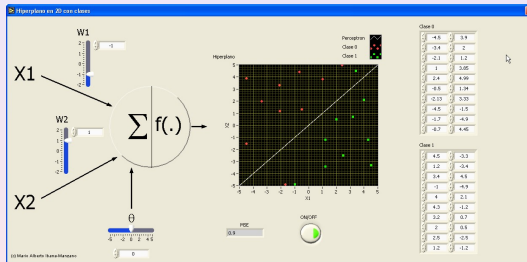


Figura: Algoritmo de aprendizaje de perceptrón de capa simple.

Definición.

- Vector de entradas: $\mathbf{x} = (x_1, x_2, \dots, x_N)^T$.
- Vector de pesos: $\mathbf{w} = (w_1, w_2, \dots, w_N)$.
- Umbral o sesgo: θ .
- Función de activación: función escalón.
- Valor de la salida deseada: $d(i) (\in \{0, 1\})$.
- Ejemplos de entrenamiento: $\{(\mathbf{x}(i), d(i)); i \in I_r\}$.
- Ejemplos de validación: $\{(\mathbf{x}(i), d(i)); i \in I_t\}$.
- Factor de aprendizaje: $\eta \left(0 < \eta < \frac{1}{\|\mathbf{x}(k)\|_{\max}} \right)$.
- Factor de momentum: $\mu (0,6 < \mu < 0,9)$.
- Función de costo: SSE ó MSE.

Descenso del gradiente.

$$E_{SSE} = \sum_{k=1}^K [e(k)]^2 = \sum_{k=1}^K [d(k) - y(k)]^2 \quad (3)$$

$$\frac{\partial E}{\partial w_i} = \frac{\partial E}{\partial e(k)} \frac{\partial e(k)}{\partial y(k)} \frac{\partial y(k)}{\partial u(k)} \frac{\partial u(k)}{\partial w_i} \quad (4)$$

$$\frac{\partial E}{\partial e(k)} = \sum_{k=1}^K \frac{\partial [e(k)]^2}{\partial e(k)} = 2 \sum_{k=1}^K e(k) = 2 \sum_{k=1}^K [d(k) - y(k)] \quad (5)$$

$$\frac{\partial e(k)}{\partial y(k)} = \frac{\partial [d(k) - y(k)]}{\partial y(k)} = -1 \quad (6)$$



Descenso del gradiente.

$$\frac{\partial y(k)}{\partial u(k)} = \frac{\partial f(u(k))}{\partial u(k)} = 1 \quad (7)$$

$$\frac{\partial u(k)}{\partial w_i} = \frac{\partial}{\partial w_i} \left(\sum_{j=1}^N w_j \cdot x_j + \theta \right) = \frac{\partial w_i \cdot x_i(k)}{\partial w_i} = x_i(k) \quad (8)$$

$$\frac{\partial E}{\partial w_i} = -2 \sum_{k=1}^K [d(k) - y(k)] x_i(k) \quad (9)$$

$$\Delta \mathbf{w}(t) = \eta \sum_{k=1}^K [d(k) - y(k)] \mathbf{x}(k) \quad (10)$$



Descenso del gradiente y momentum.

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \Delta \mathbf{w}(t) \quad (11)$$

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \eta \sum_{k=1}^K \{[d(k) - y(k)] \cdot \mathbf{x}(k)\} \quad (12)$$

$$\mathbf{w}(t+1) = \Delta \mathbf{w}(t) + \mu [\mathbf{w}(t) - \mathbf{w}(t-1)] \quad (13)$$



Algoritmo de entrenamiento.

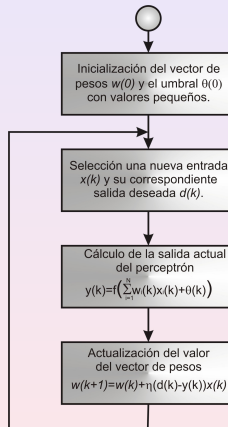


Figura: Algoritmo de aprendizaje para el perceptrón.

Algoritmo de entrenamiento del perceptrón simple.

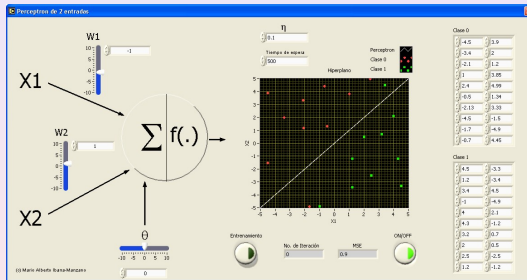


Figura: Algoritmo de aprendizaje de perceptrón de capa simple.

Algoritmo de entrenamiento del perceptrón simple.

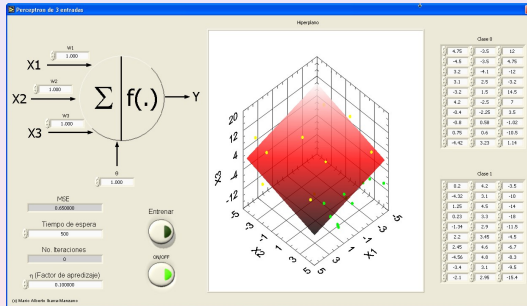


Figura: Algoritmo de aprendizaje de perceptrón de capa simple.

Aplicaciones del perceptrón.

- La transformación no lineal del vector de características, no es específica del problema.
- El algoritmo de aprendizaje del perceptrón no converge para valores fijos del factor de aprendizaje η , si los patrones de entrenamiento no son linealmente separables.
- Si el patrón de entrada es linealmente separable, entonces no se conoce el tiempo que le tomará al algoritmo de entrenamiento el converger al vector de pesos que corresponda al hiperplano necesario para separar los patrones.

