Universidad Autónoma del Estado de Hidalgo

Escuela Superior Huejutla

Área Académica: Instituto de Ciencias Básicas e Ingeniería, Sistemas Computacionales

Tema: Perceptron Parte I

Profesor: Víctor Tomás T. Mariano.

Alumnos:

Leticia Hernández Hernández

Agustín Hernández Espinoza

Periodo: Julio-Diciembre 2011

Resumen:

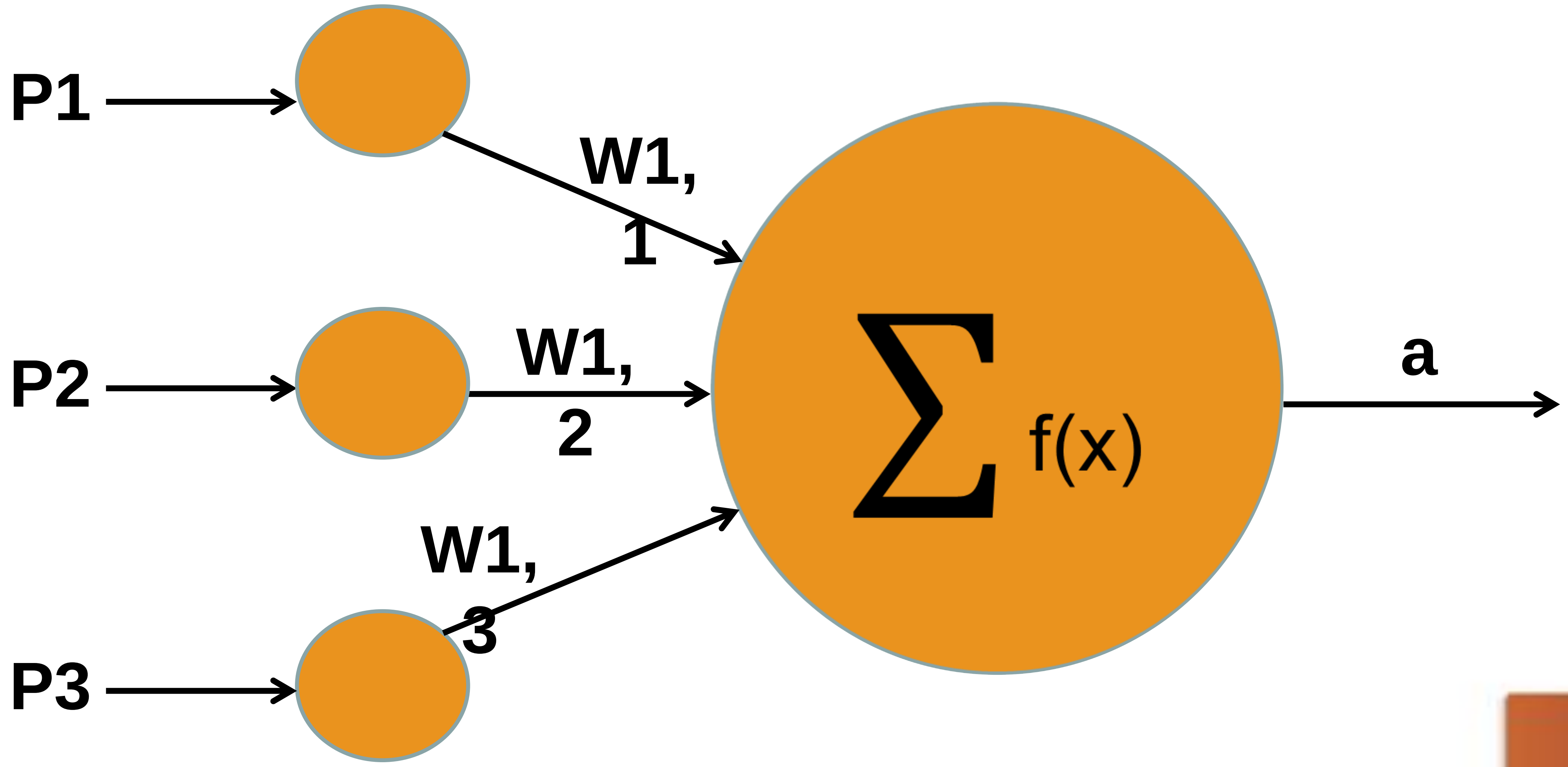
Se describe el modelo del Perceptrón, historia, características, algoritmo de entrenamiento para resolver problemas de clasificación, se presentan también ejemplos resueltos, así como las limitantes de este tipo de red.

Abstract

We describe the Perceptron model, history, characteristics, training algorithm for solving classification problems, worked examples are also presented, as well as the limitations of this type of network.

Keywords: Perceptrón, rule training.

EL PERCEPTRÓN



HISTORIA

En 1943, Warren McCulloch y Walter Pitts introdujeron una de las primeras neuronas artificiales. La característica principal de su modelo de neurona es que un suma ponderada de las señales de entrada se compara con un umbral para determinar la neurona de salida. Cuando la suma es mayor o igual al umbral, la salida es 1. Cuando la suma es menor que el umbral, la salida es 0.

A finales de 1950 Frank Rosenblatt y otros investigadores desarrollaron una clase de redes neuronales llamadas perceptrones. Las neuronas de estas redes eran similares a las de McCulloch y Pitts.

HISTORIA

La contribución clave de Rosenblatt fue la introducción de una regla de aprendizaje para la formación de redes perceptrón para resolver problemas de reconocimiento de patrones. Demostró que su regla de aprendizaje siempre convergirá a los pesos correctos de la red, si existen pesos que solucionan el problema. El Perceptrón pudo incluso aprender cuando se inicializaba con valores aleatorios de sus pesos y bias.

HISTORIA

El Perceptrón es limitado. Dichas limitaciones fueron publicadas en el libro Perceptrons por Marvin Minsky y Seymour Papert. Ellos demostraron que las redes perceptrón eran incapaces de implementar ciertas funciones elementales. No fue sino hasta la década de los 8O's que estas limitaciones fueron superadas con las redes perceptrón mejoradas (multicapa) asociadas con reglas de aprendizaje.

Reglas de Aprendizaje

Por Reglas de Aprendizaje nos referimos a un procedimiento para modificar los pesos y biases de una red (también conocido como algoritmo de entrenamiento). El propósito de la Regla de Aprendizaje es entrenar la red para realizar alguna tarea. Existen varios tipos de reglas de aprendizaje de redes neuronales. Se dividen en tres categorías: Aprendizaje Supervisado, Aprendizaje No Supervisado y Aprendizaje por Reforzamiento.

Aprendizaje Supervisado

En este tipo de aprendizaje, la regla de aprendizaje cuenta con un conjunto de ejemplos (conjunto de entrenamiento) de comportamiento de la red.

$$\{p_1, t_1\}, \{p_2, t_2\}, \ldots, \{p_Q, t_Q\}$$

Donde p_q es una entrada a la red y t_q corresponde a la salida correcta (objetivo). Como las entradas se aplican a la red, la salida de la red se compara con los objetivos. La regla de aprendizaje se utiliza para ajustar los pesos y biases de la red con el fin de mover las salidas de la red mas cerca de los objetivos. La regla de aprendizaje del Perceptrón cae en esta categoría de Aprendizaje Supervisado.

Arquitectura del Perceptrón

La red general del Perceptrón se muestra en la sig. Figura:

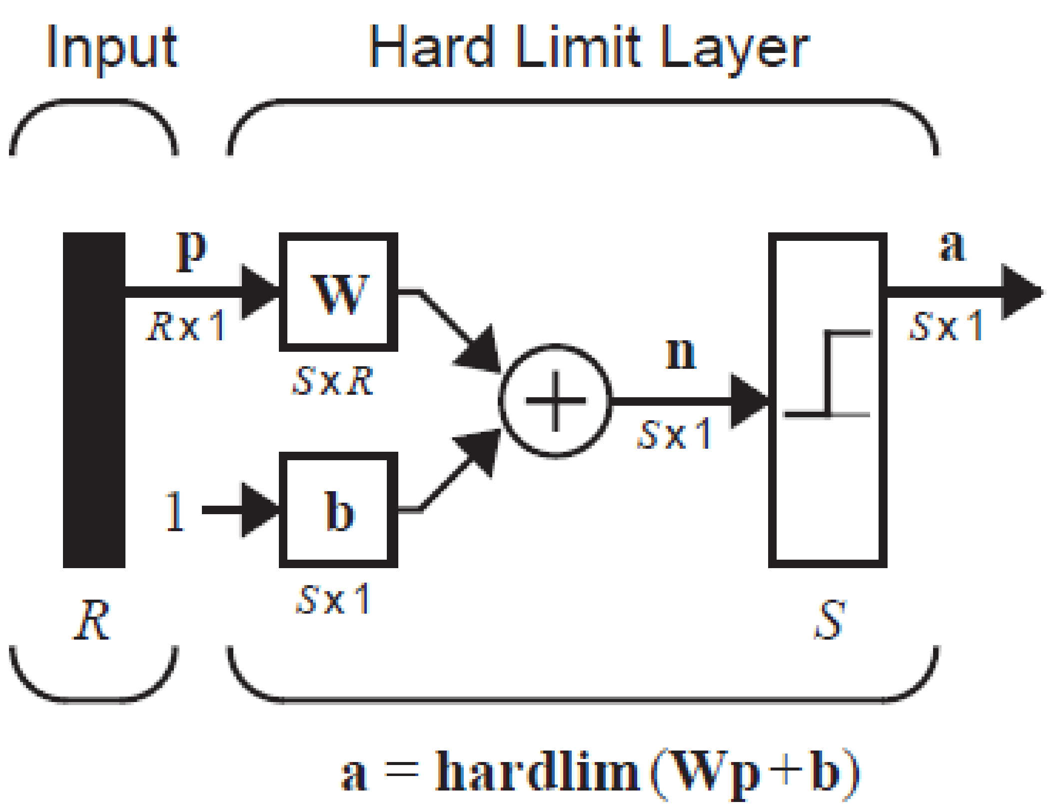


Figura 4.1 Red Perceptrón

Arquitectura del Perceptrón

La salida de la red esta dada por:

$$a = hardlim(Wp + b)$$

Esto nos será útil en el desarrollo de la regla de aprendizaje del Perceptrón para ser capaz de hacer referencia de los elementos individuales de la salida de la red. Primero, consideremos la matriz de pesos de la red:

$$W = \begin{bmatrix} w_{1,1} & w_{1,2} \cdots & w_{1,R} \\ w_{2,1} & w_{2,2} \cdots & w_{2,R} \\ \vdots & \vdots & \vdots \\ w_{S,1} & w_{S,2} & w_{S,R} \end{bmatrix}$$

Arquitectura del Perceptrón

Definiremos un vector compuesto de los elementos de la i-ésima fila de W:

$$ {}_{i}W = \begin{bmatrix} w_{i,1} \\ w_{i,2} \\ \vdots \\ w_{i,R} \end{bmatrix} $$

Ahora podemos dividir la matriz de pesos:

$$ W = \begin{bmatrix} {}_{1}w^{T} \\ {}_{2}w^{T} \\ \vdots \\ {}_{S}w^{T} \end{bmatrix} $$

Esto nos permite escribir el i- ésimo elemento del vector de salida de la red.

Arquitectura del Perceptrón

$$a_i = hardlim(n_i) = \mathrm{hardlim}\left({}_i w^T \mathrm{p} + b_i\right).$$

Recordemos que la función de transferencia hardlim esta definida por:

$$\text{a=hardlim(n)} \begin{cases} 1 \; si \; n \geq 0 \\ 0 \; si \; n < 0 \end{cases}$$

Por lo tanto si el producto de la i-esima fila de la matriz de pesos y el vector de entrada es mayor o igual a $-b_i$ es igual a 1, en caso contrario la salida es 0. Asi cada neurona en la red divide el espacio de entrada en dos regiones. Esto es util para investigar los limites entre estas regiones.

Perceptrón de una sola neurona

Consideremos un Perceptrón con una neurona de dos entradas, como se muestra en la figura:

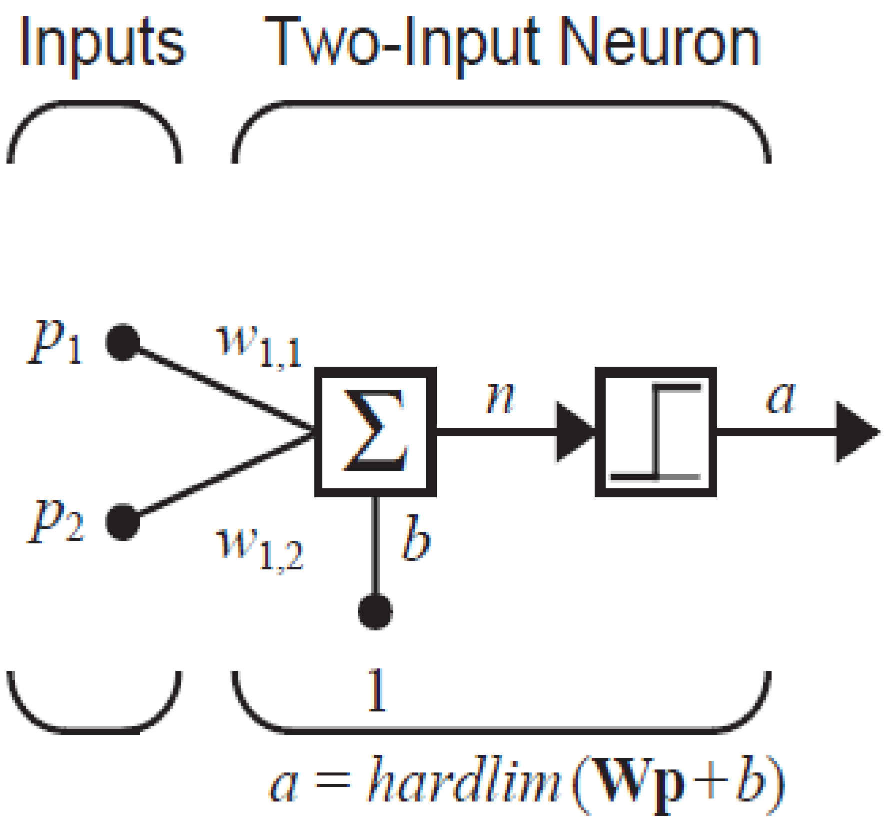


Figura 4.2 Perceptrón de dos entradas/una sola salida.

Perceptrón de una sola neurona

La salida de esta red está determinada por:

$$a = hardlim(n) = hardlim(Wp + b)$$
$$= hardlim\left({}_1w^T p + b\right) = hardlim(w_{1,1} p_1 + w_{1,2} p_2 + b)$$

El límite de decisión está determinada por los vectores de entrada para los cuales la entrada neta n es cero:

$$n = {}_1w^T p + b = w_{1,1} p_1 + w_{1,2} p_2 + b = 0$$

Ejemplo: asignaremos los siguientes valores a los pesos y bias:

$$w_{1,1} = 1, \qquad w_{1,2} = 1, \qquad b = -1$$

El límite de decisión es entonces:

$$n = {}_1w^T p + b = w_{1,1} p_1 + w_{1,2} p_2 + b = p_1 + p_2 - 1 = 0$$

Limite de decisión

Esto define una línea sobre el espacio de entrada. Sobre un lado de la línea la salida de la red seria 0. Sobre la línea y sobre el otro lado de la línea la salida seria 1. Parea dibujar la línea podemos encontrar los puntos donde se intersectan los ejes p1 y p2. Para encontrar intersección p2 colocamos $p_1 = 0$.

$$p_2 = -\frac{b}{w_{1,2}} = \frac{-1}{1} = 1 \; si \; p_1 = 0$$

Para encontrar intersección p1 colocamos p2=0:

$$p_1 = -\frac{b}{w_{1,1}} = \frac{-1}{1} = 1 \; si \; p_1 = 0$$

Limite de decisión

El resultado se muestra en la siguiente figura:

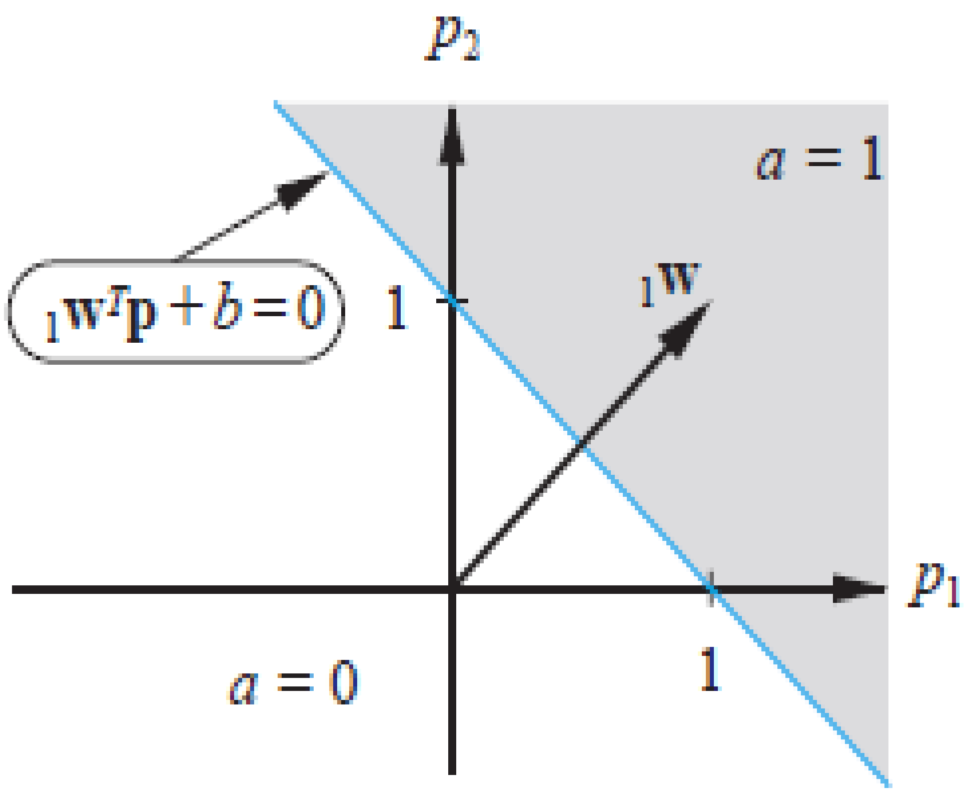


Figura 4.3 Limite de decisión para Perceptrón de dos entradas

Limite de decisión

Para saber que lado del limite corresponde a una salida de 1, solo se necesita probar con un punto. Para la entrada $p = [2\ 0]^T$ la salida de la red será:

$$a = {}_1w^T p + b = hardlim\left([1\ 1]\begin{bmatrix}2\\0\end{bmatrix} - 1\right) = 1$$

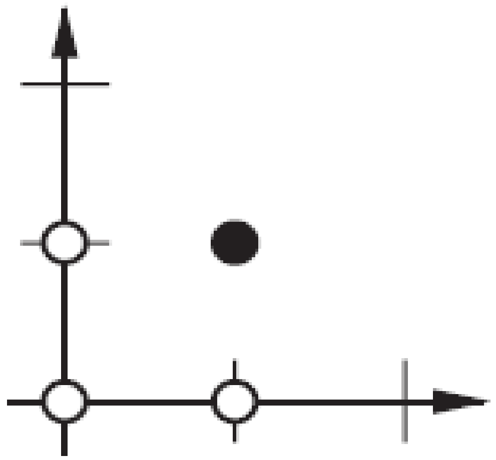
Por lo tanto la salida de la red será 1 del lado derecho de la parte superior del limite de decisión. Esta región se indica por el área sombreada en la figura 4.3

Limite de decisión: ejemplo

Aplicando algunos de los conceptos del diseño de una red perceptrón implementando una función lógica simple: la puerta AND. Los pares entrada/objetivo para la puerta AND son:

$$\left\{p_1 = \begin{bmatrix}0\\0\end{bmatrix}, t_1 = 0\right\}\left\{p_2 = \begin{bmatrix}0\\1\end{bmatrix}, t_2 = 0\right\}\left\{p_3 = \begin{bmatrix}1\\0\end{bmatrix}, t_3 = 0\right\}\left\{p_4 = \begin{bmatrix}1\\1\end{bmatrix}, t_4 = 1\right\}$$

En la siguiente figura se muestra el problema gráficamente:



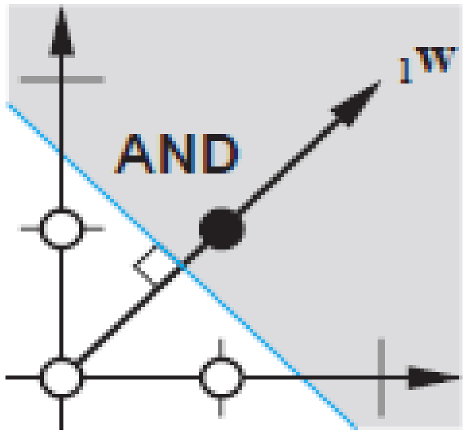
Los círculos negros indican que el objetivo es 1, y los círculos blancos indican que el objetivo es 0.

Limite de decisión: ejemplo

El primer paso del diseño consiste en seleccionar un límite de decisión. Queremos tener una línea que separe los círculos obscuros de los círculos en blanco. Parece razonable elegir la línea ”intermedia” que cae entre las dos categorías de entradas.

Después elegiremos un vector de pesos que es ortogonal al limite de decisión. El vector de pesos puede ser de cualquier longitud, entonces hay posibilidades infinitas. Una de ellas es:

$$_1w = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$



Limite de decisión: ejemplo

Finalmente, necesitamos encontrar el bias. Esto lo podemos hacer eligiendo un punto en el limite de decisión y satisfaciendo la ecuación:

${}_1w^T p + b = 0$. Si usamos $p = [1.5\ 0]^T$ encontramos:

$${}_1w^T p + b = [2\ 2]\begin{bmatrix}1.5\\0\end{bmatrix} + b = 3 + b = 0 \qquad \rightarrow b = -3$$

Ahora podemos probar la red en un de los pares entrada/objetivo. Si aplicamos $p_2 = \begin{bmatrix}0\\1\end{bmatrix}$ a la red, la salida será:

$$a = hardlim({}_1w^T p_2 + b) = hardlim\left([2\ 2]\begin{bmatrix}0\\1\end{bmatrix} - 3\right)$$

$$a = hardlim(-1) = 0$$

La cual es igual a la salida objetivo $t_2 = 0$.

Verificar que todas las entradas son correctamente clasificadas.

Perceptrón de Múltiples Neuronas

Para Perceptrones con multiples neuronas habría un limite de decisión por cada neurona. El limite de decisión para una neurona i esta definida por:

$$ {}_{i}w^{T} p + b_{i} = 0 $$

Un Perceptrón con una sola neurona puede clasificar vectores de entrada en dos categorías, dado que su salida puede ser cualquiera de las dos: 0 ó 1. Un Perceptrón de múltiples neuronas puede clasificar entradas dentro de muchas categorías. Cada categoría es representada por un vector de salida diferente. Ya que cada elemento del vector de entrada puede ser 0 ó 1, hay un total de 2^{S} posibles categorías, donde S es el numero de neuronas.

Regla de Aprendizaje del Perceptrón

Esta regla de aprendizaje es un ejemplo de Aprendizaje Supervisado, en el cual la regla de aprendizaje esta provista con un conjunto de ejemplos de propiedades de comportamiento de una red:

$$\{p_1, t_1\}, \{p_2, t_2\}, \ldots, \{p_Q, t_Q\},$$

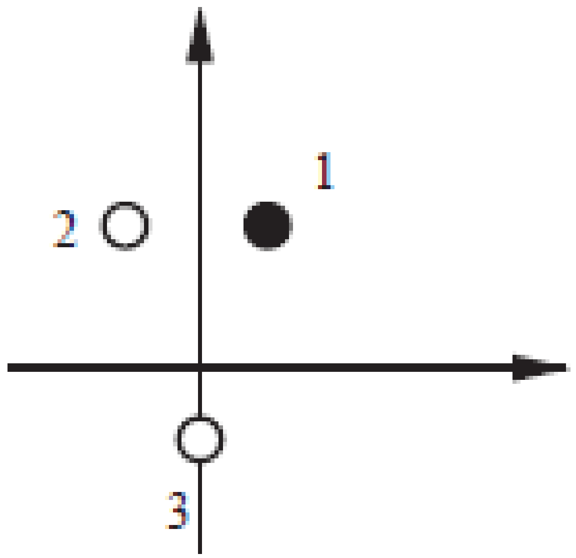
Donde p_q es una entrada a la red y t_q es la salida objetivo correspondiente. Como cada entrada se aplica a la red, la salida de la red se compara con el objetivo. Como las entradas se aplican a la red, la salida de la red se compara con los objetivos. La regla de aprendizaje se utiliza para ajustar los pesos y biases de la red con el fin de mover las salidas de la red mas cerca de los objetivos.

Regla de Aprendizaje del Perceptrón: Problema de prueba

Los pares de entrada / objetivo para nuestro problema de prueba son:

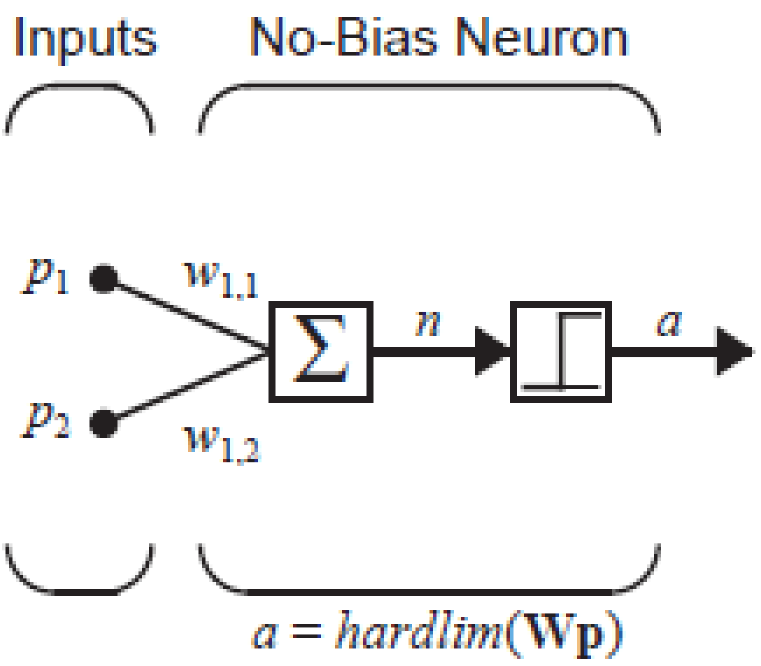
$$\left\{p_1 = \begin{bmatrix}1\\2\end{bmatrix}, t_1 = 1\right\} \left\{p_2 = \begin{bmatrix}-1\\2\end{bmatrix}, t_2 = 0\right\} \left\{p_3 = \begin{bmatrix}0\\-1\end{bmatrix}, t_3 = 0\right\}$$

El problema se muestra gráficamente; los dos vectores de entrada, uno cuyo objetivo es 0 (círculos blancos), y otro cuyo objetivo es 1 (círculos oscuros).



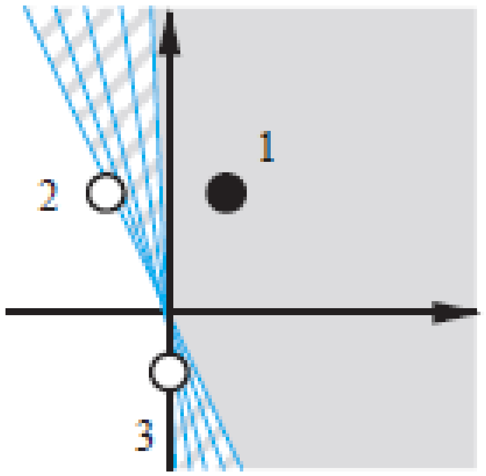
Regla de Aprendizaje del Perceptrón: Problema de prueba

La red de este problema debería tener dos entradas y una salida. Para simplificar el desarrollo de la regla de aprendizaje, empezaremos con una red sin un bias. La red tendrá solo dos parámetros: $w_{1,1}$ y $w_{1,2}$.



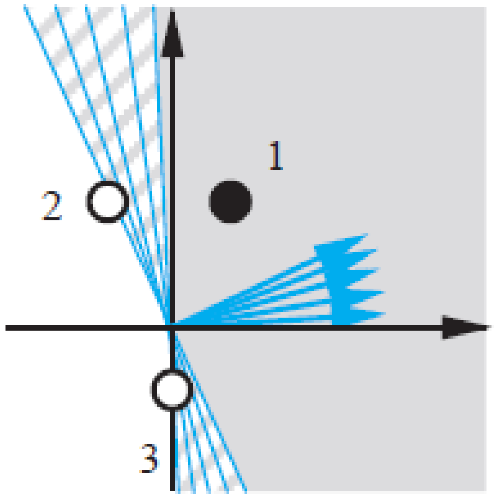
Regla de Aprendizaje del Perceptrón: Problema de prueba

Al eliminar el bias nos quedamos con una red cuyo limite de decisión debería pasar por el origen. Tiene que haber un admisible limite de decisión que separe los vectores p_2y p_3 del vector p_1. La figura muestra que existe un numero infinito de tales limites.



Regla de Aprendizaje del Perceptrón: Problema de prueba

La siguiente figura muestra los vectores de pesos que corresponden a los limites de decisión. Nos gustaría una regla de aprendizaje que encuentre un vector de pesos que apunte en una de estas direcciones. La longitud del vector de pesos no importa, sólo su dirección es importante.



Bibliografía

Neural Network Design. [Martin T. Hagan, Howard B, Demuth, Mark Beale – PWS Publishing Company].