

# TAREA 1: DISEÑO DE PERCEPTRON EN EXCEL

Marcos Rafael Roman Salgado<sup>1</sup>

<sup>1</sup>UACH, Texcoco de Mora – <marcosrafael2000@hotmail.com>

## RESUMEN

En esta práctica, implementaremos un perceptrón en Excel para entender su funcionamiento. Utilizaremos tres entradas binarias (0 o 1) y ajustaremos los pesos y el sesgo para entrenar el modelo. En cada iteración, calcularemos la salida mediante la función de activación y compararemos esta salida con los valores esperados para ajustar los pesos. Esta práctica nos permitirá visualizar el proceso de aprendizaje del perceptrón y comprender cómo los ajustes en los pesos afectan la precisión del modelo. Esta experiencia fortalecerá nuestro conocimiento sobre los fundamentos de las redes neuronales y su aplicación en problemas de clasificación.

**Palabras-clave:** *perceptron, redes neuronales, excel, ajuste de peso*

## ABSTRACT

In this practice, we will implement a perceptron in Excel to understand its function. Using three binary inputs (0 or 1), we will adjust weights and bias to train the model. Each iteration involves calculating the output via the activation function and adjusting the weights based on expected values. This exercise allows us to visualize the perceptron's learning process and understand how weight adjustments impact model accuracy, enhancing our grasp of neural network fundamentals and classification applications.

**Keywords:** *perceptron, neural networks, excel, weight adjustment*

## 1. INTRODUCCIÓN

La inteligencia artificial y el aprendizaje automático son campos en rápido crecimiento con aplicaciones en diversas áreas. Un componente fundamental de estos campos es el perceptrón, una de las estructuras más simples de las redes neurona-

les. Esta práctica tiene como objetivo proporcionar una comprensión práctica y teórica del perceptrón utilizando una herramienta accesible y común: Microsoft Excel.

### Atencion

- **Preparación de Datos:** Asegúrate de tener listas las entradas binarias (0 o 1) antes de comenzar.
- **Cálculos Iterativos:** Seguiremos un proceso iterativo para ajustar los pesos y el sesgo del perceptrón.
- **Visualización del Proceso:** Tablas en Excel para visualizar el proceso de aprendizaje del perceptrón.
- **Al final de la práctica,** analizaremos cómo los ajustes en los pesos afectan la precisión del modelo.

## 2. MATERIALES Y MÉTODOS

### 2.1. Materiales

- Software: Microsoft Excel
- Hardware: Computadora
- Datos:
  - Entradas Binarias
  - Salidas esperadas

### 2.2. Métodos

1. Preparación de datos:
  - Crear una tabla de excel con las tres entradas binarias y salidas esperadas.
2. Inicialización del perceptrón:
  - Asignar valores de 0 a los pesos.
  - Insertar los valores en celdas específicas de Excel.
3. Cálculo de salida:
  - Utilizar la función de activación para calcular la salida del perceptrón.
  - Implementar la formula en Excel usando las celdas correspondientes.

#### 4. Calculo del error y ajuste de pesos:

- Calcular el error comparando la salida del perceptrón con la salida esperada.
- Ajustar los pesos utilizando la regla de aprendizaje del perceptrón.
- implementar estas formulas en Excel y repetir el proceso para varias iteraciones.

#### 5. Iteraciones:

- Realizar múltiples iteraciones ajustando los pesos y el sesgo hasta que el error sea minimizado o se alcance un número predefinido de iteraciones.
- Visualizar cómo cambian los pesos y la precisión del modelo a lo largo de las iteraciones.

#### 6. Análisis de Resultados

- o Evaluar la precisión final del modelo comparando las salidas calculadas con las salidas esperadas.

### 3. RESULTADOS

#### 3.1. Ajuste de pesos

Durante la práctica, se observaron múltiples iteraciones en las que los pesos y el sesgo del perceptrón se ajustaron con base en el error de predicción.<sup>1</sup>

#### 3.2. Salidas Calculadas y Precisión

Se compararon las salidas calculadas del perceptrón con las salidas esperadas a lo largo de varias iteraciones para evaluar la precisión del modelo. La siguiente tabla muestra un resumen de las salidas esperadas y las salidas calculadas al final del proceso de entrenamiento:

In 1	In 2	In 3	Esperada	Obtenida
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0

**Cuadro 1.** Tabla de entradas, salidas esperadas y obtenidas.

#### 3.3. Visualización del Proceso de Aprendizaje

Se utilizaron gráficos en Excel para visualizar la evolución de los pesos del perceptrón a lo largo de las iteraciones del entrenamiento. Estos gráficos proporcionaron una representación clara de cómo los pesos se ajustaban y estabilizaban con cada iteración. Inicialmente, los pesos experimentaron cambios significativos mientras el perceptrón ajustaba sus parámetros en respuesta a los errores de predicción. Con el avance de las iteraciones, se observó una tendencia a la estabilización de los pesos, indicando que el modelo estaba convergiendo a una

**Figura 1.** Tabla completa de valores.

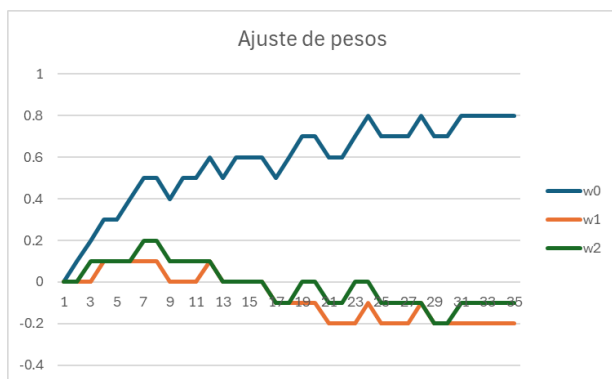
Entrada				Pesos iniciales			Salida					Error	Corrección	Pesos finales		
Valores de sensor				Salida deseada			Sensor			Suma	Red					
$x_0$	$x_1$	$x_2$	$z$	$w_0$	$w_1$	$w_2$	$c_0$	$c_1$	$c_2$	$s$	$n$	$e$	$d$	$w_0$	$w_1$	$w_2$
							$x_0 * w_0$	$x_1 * w_1$	$x_2 * w_2$	$c_0 + c_1 + c_2$	if $s > t$ then 1, else 0	$z - n$	$r * e$	$\Delta(x_0 * d)$	$\Delta(x_1 * d)$	$\Delta(x_2 * d)$
1	0	0	1	0	0	0	0	0	0	0	0	1	0.1	0.1	0	0
1	0	1	1	0.1	0	0	0.1	0	0	0	0	1	0.1	0.2	0	0.1
1	1	0	1	0.2	0	0.1	0.2	0	0	0.2	0	1	0.1	0.3	0.1	0.1
1	1	1	0	0.3	0.1	0.1	0.3	0.1	0.1	0.5	0	0	0	0.3	0.1	0.1
1	0	0	1	0.3	0.1	0.1	0.3	0	0	0.30	0	1	0.1	0.4	0.1	0.1
1	0	1	1	0.4	0.1	0.1	0.4	0	0.1	0.50	0	1	0.1	0.5	0.1	0.2
1	1	0	1	0.5	0.1	0.2	0.5	0.1	0	0.60	1	0	0	0.5	0.1	0.2
1	1	1	0	0.5	0.1	0.2	0.5	0.1	0.2	0.80	1	-1	-0.1	0.4	0	0.1
1	0	0	1	0.4	0	0.1	0.4	0	0	0.40	0	1	0.1	0.5	0	0.1
1	0	1	1	0.5	0	0.1	0.5	0	0.1	0.60	1	0	0	0.5	0	0.1
1	1	0	1	0.5	0	0.1	0.5	0	0	0.50	0	1	0.1	0.6	0.1	0.1
1	1	1	0	0.6	0.1	0.1	0.6	0.1	0.1	0.80	1	-1	-0.1	0.5	0	0
1	0	0	1	0.5	0	0	0.5	0	0	0.50	0	1	0.1	0.6	0	0
1	0	1	1	0.6	0	0	0.6	0	0	0.60	1	0	0	0.6	0	0
1	1	0	1	0.6	0	0	0.6	0	0	0.60	1	0	0	0.6	0	0
1	1	1	0	0.6	0	0	0.6	0	0	0.60	1	-1	-0.1	0.5	-0.1	-0.1

Fuente: Autoria propia.

solución más óptima.

Los gráficos de pesos mostraron cómo cada peso individual ( $w_0$ ,  $w_1$ ,  $w_2$ ) variaba en función del número de iteraciones. Se pudo observar que, tras un número suficiente de iteraciones, los cambios en los pesos se redujeron, lo que reflejó una disminución en el error total del modelo. Esta estabilización de los pesos es indicativa de que el perceptrón había aprendido a clasificar correctamente las entradas en función de las salidas esperadas.

**Figura 2.** Proceso de aprendizaje.



Fuente: Autoría propia.

### 3.4. Resultados de los Pesos Finales

A continuación se presentan los valores finales de los pesos y el sesgo después de completar las iteraciones:

$w_0$	$w_1$	$w_2$
0.8	-0.2	-0.1

**Cuadro 2.** Valores de pesos finales

## 4. DISCUSIÓN

El proceso de ajuste de pesos y sesgo en el perceptrón fue efectivo para minimizar el error total y mejorar la precisión del modelo. Al final de las iteraciones, el perceptrón fue capaz de clasificar correctamente todas las muestras del conjunto de datos. Esta práctica demostró cómo los principios básicos de los perceptrones y el aprendizaje supervisado pueden ser implementados y visualizados utilizando una herramienta accesible como Excel.

## 5. CONCLUSIONES

- **Comprensión Práctica:** La práctica proporcionó una comprensión práctica del funcionamiento

de un perceptrón.

- **Visualización del Aprendizaje:** Excel permitió una visualización clara del proceso de aprendizaje del perceptrón.
- **Fundamentos de Redes Neuronales:** Los estudiantes adquirieron conocimientos fundamentales sobre redes neuronales y clasificación binaria.

Esta experiencia práctica ha fortalecido el entendimiento teórico y práctico de los estudiantes, preparándolos mejor para abordar problemas más complejos en el campo del aprendizaje automático y las redes neuronales.

## APÉNDICES

### A. FORMULAS UTILIZADAS EN EXCEL

Para el cálculo de los pesos en Excel, se utilizó la siguiente fórmula:

$$\text{Nuevo Peso} = \text{Peso Actual} + (\text{Aprendizaje} \times \text{Entrada}) \quad (1)$$

Donde:

- **Peso Actual** corresponde al valor del peso antes de la actualización.
- **Tasa de Aprendizaje** es un parámetro que controla el tamaño del ajuste del peso.
- **Entrada** es el valor de la entrada correspondiente.

Si **Peso Actual** está en la celda E18, **Tasa de Aprendizaje** en la celda N18, y **Entrada** en la celda A18, entonces la fórmula en Excel es:

$$\text{Nuevo Peso} = E18 + (N18 \times A18) \quad (2)$$

### ■ Contacto

Para cualquier duda favor de contactar:

✉ [marcosrafael2000@hotmail.com](mailto:marcosrafael2000@hotmail.com)

☎ +52 5537149673

### ■ Referencias

- I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.
- C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

F. Rosenblatt, "The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain," *Psychological Review*, vol. 65, no. 6, pp. 386-408, 1958.

M. Minsky and S. Papert, *Perceptrons: An Introduction to Computational Geometry*, MIT Press, 1969.