

UNIVERSIDAD DE GUADALAJARA



CENTRO UNIVERSITARIO DE CIENCIAS
EXACTAS E INGENIERÍA
DEPARTAMENTO DE
CIENCIAS
COMPUTACIONALES

Seminario de Solución de Problemas de Sistemas Basados en Conocimiento

Práctica No. 10

Perceptrón simple: Compuertas Lógicas “OR” y “AND”

Nombre: Hurtado González Edgar Arturo
Código: 212597894

Introducción

El perceptrón simple es un modelo de red neuronal de una sola capa. Fue propuesto por Frank Rosenblatt en 1957 y es uno de los primeros tipos de redes neuronales desarrollados.

El perceptrón simple consiste en un conjunto de entradas x_1, x_2, \dots, x_n , cada una multiplicada por un peso correspondiente w_1, w_2, \dots, w_n . Estos productos se suman y se pasa el resultado a través de una función de activación. La salida final es la respuesta del perceptrón después de que la suma ponderada ha sido procesada por la función de activación.

La función de activación más comúnmente usada en un perceptrón simple es la función escalón unitario, que produce una salida binaria. Si la suma ponderada de las entradas supera un cierto umbral, el perceptrón produce una salida de 1; de lo contrario, produce una salida de 0.

El perceptrón simple es capaz de aprender y ajustar sus pesos a través de un algoritmo de aprendizaje supervisado llamado regla de aprendizaje del perceptrón. Este algoritmo ajusta los pesos en función de la diferencia entre la salida calculada por el perceptrón y la salida deseada, lo que permite al perceptrón aprender a clasificar datos.

Sin embargo, el perceptrón simple tiene limitaciones. Por ejemplo, no puede resolver problemas que no son linealmente separables, lo que significa que no puede aprender patrones que no se pueden separar mediante una sola línea en el espacio de entrada. Para abordar esta limitación, se han desarrollado redes neuronales más complejas, como los perceptrones multicapa y las redes neuronales profundas, que pueden manejar problemas más complejos y realizar tareas de aprendizaje más avanzadas.

El perceptrón simple es la forma más básica de una red neuronal. Funciona tomando entradas (datos o características), multiplicándolas por pesos correspondientes y sumándolas. Esta suma ponderada se pasa a través de una función de activación que decide si el perceptrón se activa o no, es decir, si produce una salida.

Para entender mejor su funcionamiento, aquí hay algunos pasos clave:

1. Entradas y Pesos:

- **Entradas:** Son los valores de características o datos que alimentan el perceptrón. Cada una de estas entradas está asociada con un peso.
- **Pesos:** Cada entrada se multiplica por un peso respectivo. Los pesos determinan la importancia relativa de cada entrada en la salida del perceptrón. Durante el entrenamiento, estos pesos se ajustan para mejorar el rendimiento del perceptrón en una tarea específica.

2. Suma Ponderada:

- Se multiplican las entradas por sus pesos correspondientes y se suman. Matemáticamente, sería algo como:
$$z = w_1 * x_1 + w_2 * x_2 + \dots + w_n * x_n$$
, donde w son los pesos y x son las entradas.

3. Función de Activación:

- La suma ponderada z se pasa a través de una función de activación. En el caso del perceptrón simple, comúnmente se usa la función escalón unitario.
- La función escalón unitario produce una salida binaria basada en la comparación de z con un umbral. Si z es mayor que el umbral, el perceptrón produce una salida de 1; de lo contrario, produce una salida de 0.

4. Ajuste de Pesos:

- Durante el entrenamiento, los pesos se ajustan utilizando un algoritmo de aprendizaje supervisado. La regla de aprendizaje del perceptrón es un ejemplo común de este proceso, que modifica los pesos en función de la diferencia entre la salida predicha y la salida real esperada.

Limitaciones del Perceptrón Simple:

- El perceptrón simple tiene limitaciones importantes. Por ejemplo, solo puede aprender a separar clases linealmente. Si los datos no se pueden dividir mediante una sola línea, el perceptrón no podrá aprender a clasificarlos correctamente.
- Esta limitación llevó al desarrollo de perceptrones multicapa y redes neuronales más complejas, como las redes neuronales profundas, que pueden manejar problemas más complejos al aprender representaciones más complejas de los datos.

La compuerta lógica OR es una de las compuertas básicas en lógica booleana y en circuitos lógicos. Toma dos entradas booleanas (0 o 1, falso o verdadero) y produce una salida verdadera (1) si al menos una de las entradas es verdadera.

Su tabla de verdad es la siguiente:

Entrada A	Entrada B	Salida OR
0	0	0
0	1	1
1	0	1
1	1	1

En términos de operaciones lógicas:

- Si A es falso (0) y B es falso (0), la salida es falso (0).
- Si A es falso (0) y B es verdadero (1), la salida es verdadero (1).
- Si A es verdadero (1) y B es falso (0), la salida es verdadero (1).
- Si A es verdadero (1) y B es verdadero (1), la salida es verdadero (1).

En diagramas de circuitos lógicos, la compuerta OR se representa típicamente con el símbolo "+" o con el símbolo "V" invertido.

La compuerta OR es esencial en la lógica digital y en la construcción de circuitos digitales, ya que junto con otras compuertas lógicas básicas (como AND, NOT, XOR) se puede utilizar para construir circuitos más complejos, como sumadores, multiplexores, flip-flops, entre otros.

En aplicaciones de programación y computación, las operaciones lógicas tipo OR son fundamentales para las estructuras de control de flujo, evaluación de condiciones y lógica de decisiones en algoritmos y programas.

La compuerta lógica AND es otra de las compuertas básicas en lógica booleana y circuitos lógicos. Al igual que la compuerta OR, toma dos entradas booleanas (0 o 1, falso o verdadero) y produce una salida verdadera (1) solo si ambas entradas son verdaderas.

Su tabla de verdad es la siguiente:

Entrada A	Entrada B	Salida AND
0	0	0
0	1	0
1	0	0
1	1	1

En términos de operaciones lógicas:

- Si A es falso (0) y B es falso (0), la salida es falso (0).
- Si A es falso (0) y B es verdadero (1), la salida es falso (0).
- Si A es verdadero (1) y B es falso (0), la salida es falso (0).
- Solo si A es verdadero (1) y B es verdadero (1), la salida es verdadero (1).

En diagramas de circuitos lógicos, la compuerta AND se representa típicamente con el símbolo "*" o con el símbolo "^".

Al igual que la compuerta OR, la compuerta AND es fundamental en la construcción de circuitos digitales y en la lógica de programación. Se utiliza en la formación de estructuras más complejas, permitiendo la realización de operaciones lógicas y matemáticas, y es esencial en el diseño de circuitos de control, registros y múltiples dispositivos lógicos.

Desarrollo

El código está implementando un perceptrón que se entrena para resolver tanto la compuerta lógica AND como la compuerta lógica OR. Lógica del código paso a paso:

1. Definición del Perceptrón:

- Se define la clase *MyPerceptron* que representa un perceptrón.
- El perceptrón se inicializa con pesos y un sesgo aleatorio, junto con una tasa de aprendizaje.

2. *Entrenamiento para la Compuerta Lógica AND:*

- Se crea un conjunto de datos de entrada y objetivos para la compuerta lógica AND.
- Se crea una instancia de *MyPerceptron* para la compuerta lógica AND y se entrena con los datos proporcionados.
- Se prueba el perceptrón entrenado con los mismos datos de entrenamiento y se muestra la predicción para cada entrada.
- Se calcula la precisión del modelo en los datos de entrenamiento.

3. *Entrenamiento para la Compuerta Lógica OR:*

- Se crean datos de entrada y objetivos para la compuerta lógica OR.
- Se crea otra instancia de *MyPerceptron* para la compuerta lógica OR y se entrena con los nuevos datos.
- Se prueba el perceptrón entrenado con los datos de entrenamiento de la compuerta lógica OR y se muestra la predicción para cada entrada.
- Se calcula la precisión del modelo en los datos de la compuerta lógica OR.

Conclusión

En conclusión, el perceptrón simple fue el punto de partida para el desarrollo de redes neuronales más complejas, y aunque tiene limitaciones, es fundamental para comprender los fundamentos del aprendizaje de máquinas y las redes neuronales artificiales.

Las compuertas lógicas son los bloques fundamentales de la lógica digital y de los circuitos electrónicos. Operan con valores booleanos (0 y 1, falso y verdadero) y son la base sobre la cual se construyen sistemas digitales complejos, desde procesadores hasta dispositivos de almacenamiento y control.

Compuerta OR: Permite obtener una salida verdadera (1) si al menos una de las entradas es verdadera.

Compuerta AND: Produce una salida verdadera (1) solo si ambas entradas son verdaderas.

El código define un perceptrón genérico y luego lo entrena para resolver dos compuertas lógicas diferentes, evaluando su rendimiento en los datos de entrenamiento para cada compuerta.

Resultados

Prueba con datos de entrenamiento para la compuerta lógica AND:

Entrada: [0 0], Predicción: 0

Entrada: [0 1], Predicción: 0

Entrada: [1 0], Predicción: 0

Entrada: [1 1], Predicción: 1

Exactitud del Modelo en los Datos de Prueba: 100.00%

Prueba con datos de entrenamiento para la compuerta lógica OR:

Entrada: [0 0], Predicción: 0


Entrada: [0 1], Predicción: 1

Entrada: [1 0], Predicción: 1

Entrada: [1 1], Predicción: 1

Exactitud en los datos de prueba para la compuerta lógica OR: 100.00%

Referencias:

González, L. (2022, 14 septiembre). ¿Qué es el perceptrón? Perceptrón simple y multicapa.  Aprende IA. <https://aprendeia.com/que-es-el-perceptron-simple-y-multicapa/>

Latam, M. (2021, 2 mayo). Compuerta OR. Mecatrónica LATAM. <https://www.mecatronicalatam.com/es/tutoriales/electronica/compuertas-logicas/compuerta-or/>

Compuertas lógicas. (s. f.). <https://www.logicbus.com.mx/compuertas-logicas.php>

Introducing EDrawMax 10. (s. f.). Edrawsoft. <https://www.edrawsoft.com/es/logic-gate/>