

Konsep Jaringan Saraf Tiruan

Randy Cahya Wihandika, S.ST., M.Kom.

Topik

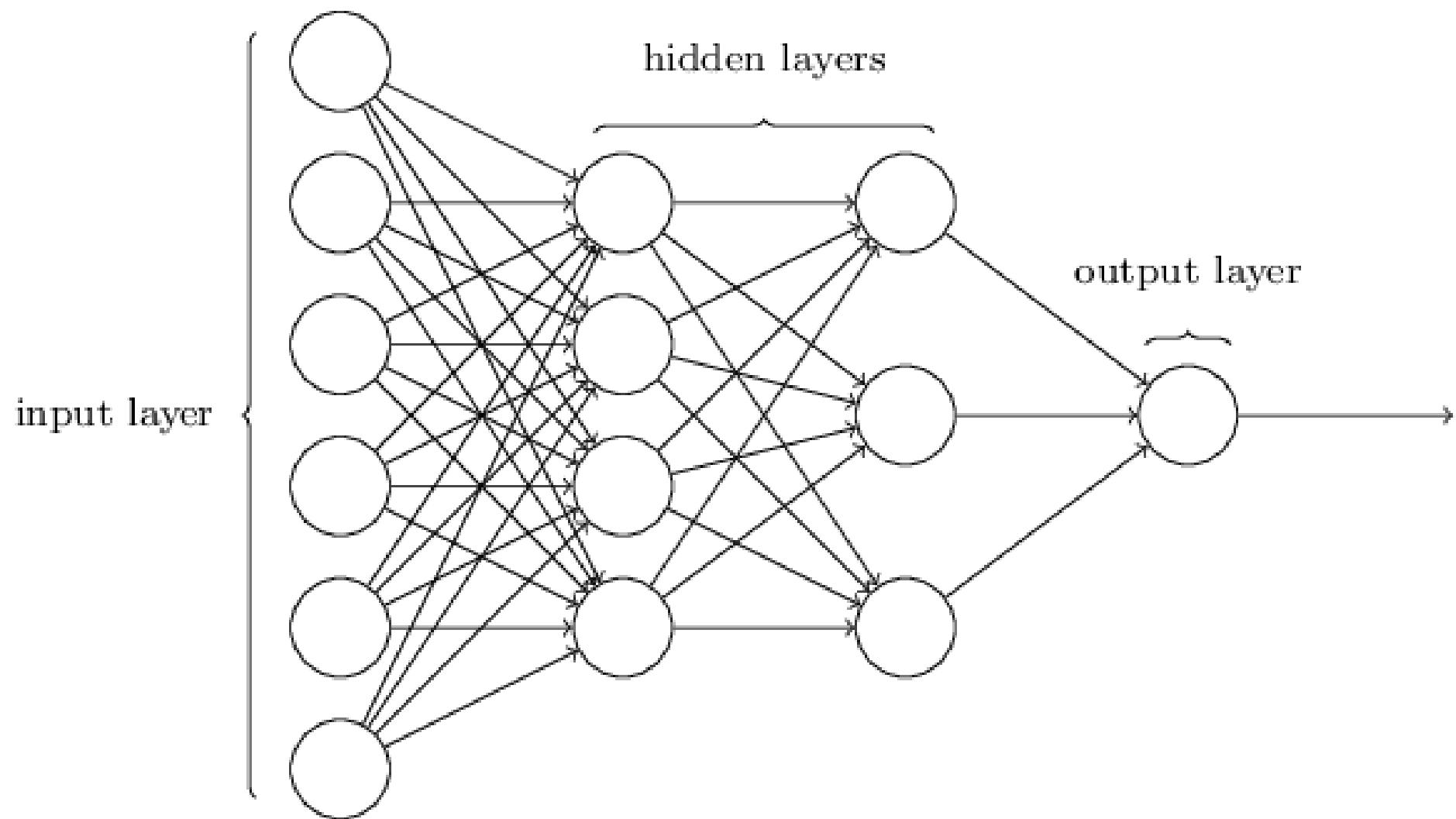
- Arsitektur JST
- Fungsi aktivasi
- McCulloch-Pitts neuron
- *Decision boundary*
- Bias

The background of the image is a dark, almost black, space. Overlaid on this are numerous glowing blue lines that represent the soma and dendrites of neurons. These lines are thin and wavy, creating a complex web-like pattern. Interspersed among these lines are several larger, more prominent neurons. One neuron on the right side has a large, rounded soma with several thick, branching processes extending outwards. Another neuron on the left side also has a large soma and is shown receiving input from other neurons. Small, bright blue dots are scattered throughout the image, representing synaptic terminals or neurotransmitter release sites.

Arsitektur JST

Arsitektur JST

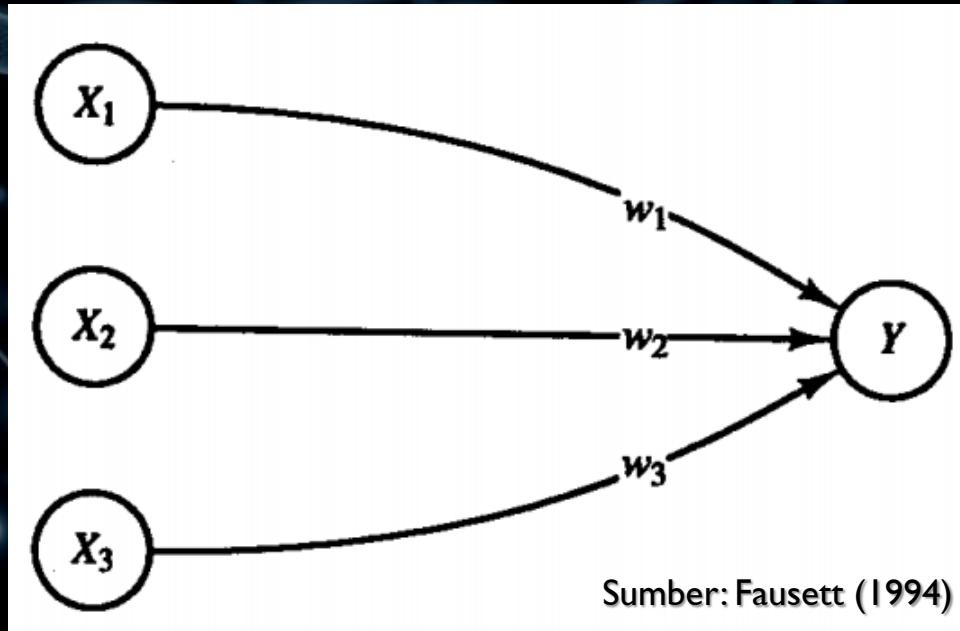
- JST umumnya divisualisasikan dalam bentuk beberapa *layer*
- Secara umum, setiap *layer* bekerja dengan cara yang sama
- Neuron di suatu layer tersambung dengan neuron-neuron di layer sebelum dan sesudahnya



Arsitektur JST

- JST dapat memiliki arsitektur *single layer* atau *multilayer*
- Umumnya, *input layer* tidak dihitung sebagai *layer* karena tidak melakukan perhitungan
- Pemrosesan data dilakukan dari *input layer* ke *output layer*
- Disebut dengan *feedforward nets*

Contoh Sederhana



$$y_{in} = x_1w_1 + x_2w_2 + x_3w_3$$

$$y_{out} = \frac{1}{1 + e^{-y_{in}}}$$



Penentuan Nilai Bobot

- y_{out} kita sebut sebagai **nilai aktivasi**
- Penentuan nilai-nilai bobot pada suatu JST dapat dilakukan secara **supervised** maupun **unsupervised**
- Supervised: klasifikasi jenis kelamin (pria 1, wanita 0)
- Unsupervised: mencari kesamaan pola pada data (*clustering*)

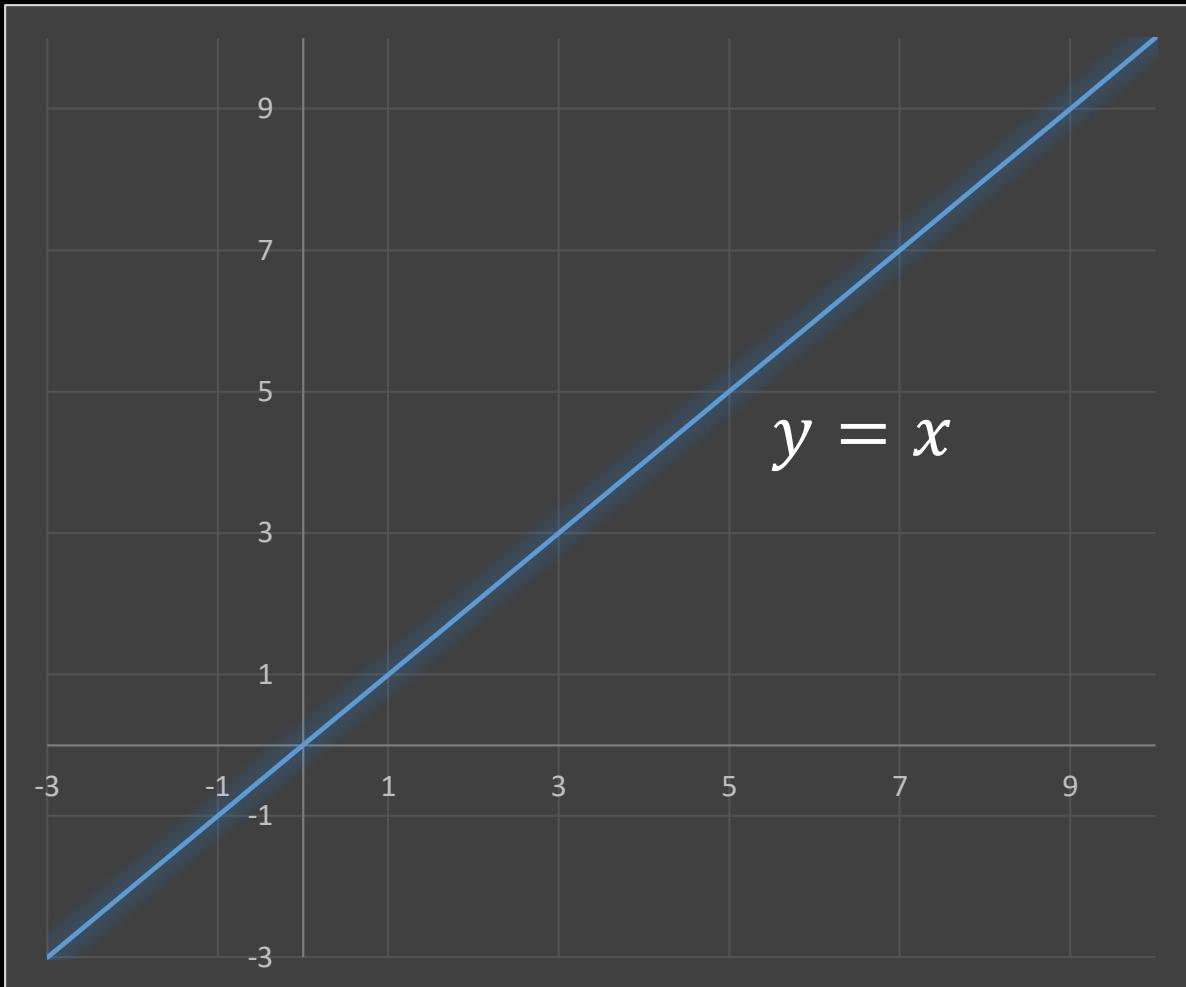
The background of the image is a dark, almost black, space. Overlaid on this are numerous thin, glowing blue lines that represent the axons and dendrites of neurons. These lines are highly branched, creating a complex web-like pattern. At various points along these lines, there are small, bright blue dots representing synaptic terminals or cell bodies. The overall effect is one of a dense, active neural network.

Fungsi Aktivasi

Fungsi Aktivasi

- Berfungsi untuk menghitung *output* dari sebuah neuron berdasarkan input yang diterimanya
- Beberapa fungsi aktivasi:
 - I. Fungsi identitas:

$$f(x) = x$$



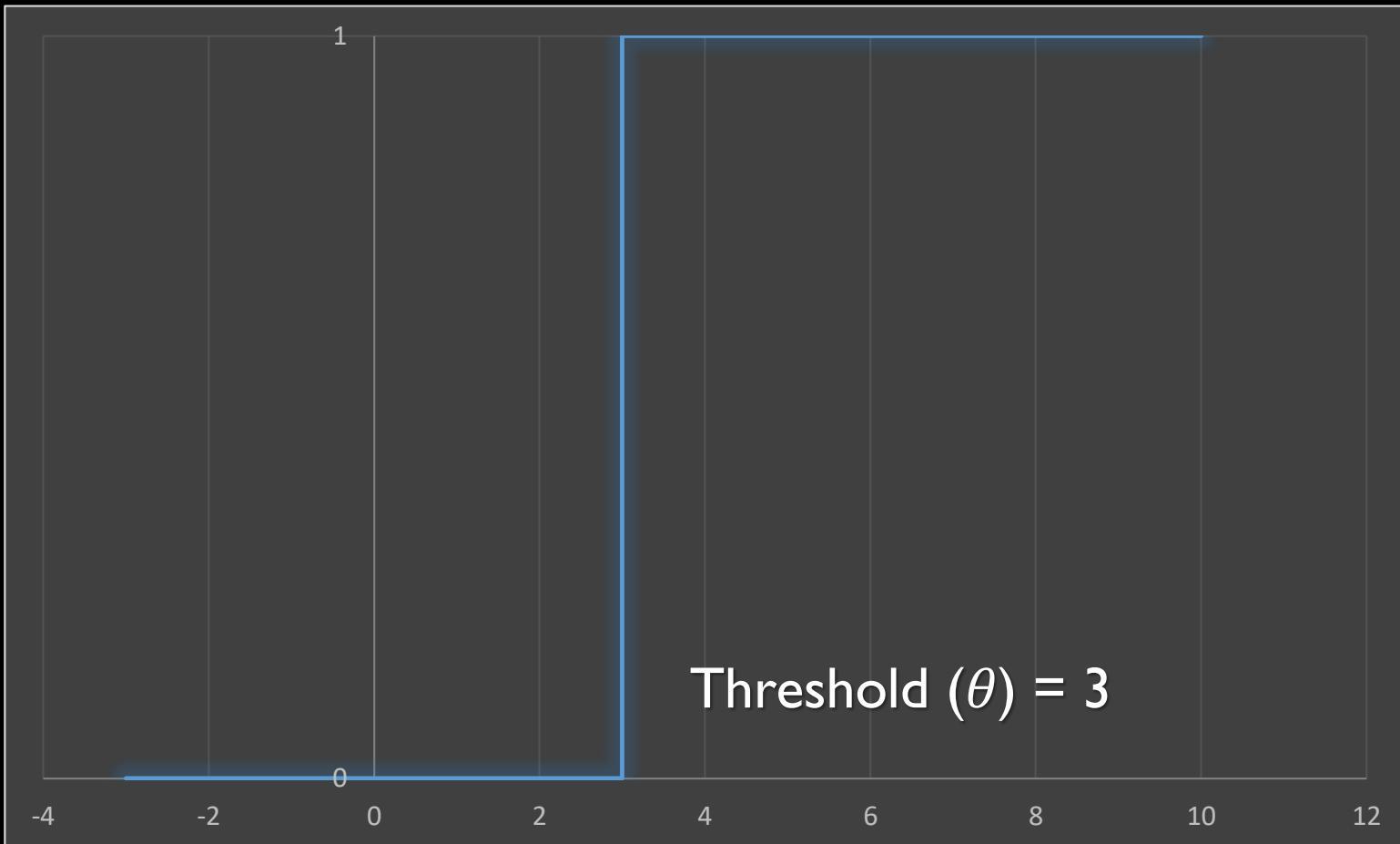
Fungsi Identitas

Fungsi Aktivasi

2. Fungsi step biner dengan nilai *threshold* θ :

$$f(x) = \begin{cases} 1 & \text{if } x \geq \theta \\ 0 & \text{if } x < \theta \end{cases}$$

- Jika nilai input suatu neuron lebih besar atau sama dengan θ , maka neuron tersebut “aktif” (menghasilkan *output* 1)
- Sebaliknya: “nonaktif” (*output* 0)



Fungsi Step

Fungsi Aktivasi

3. Fungsi step bipolar:

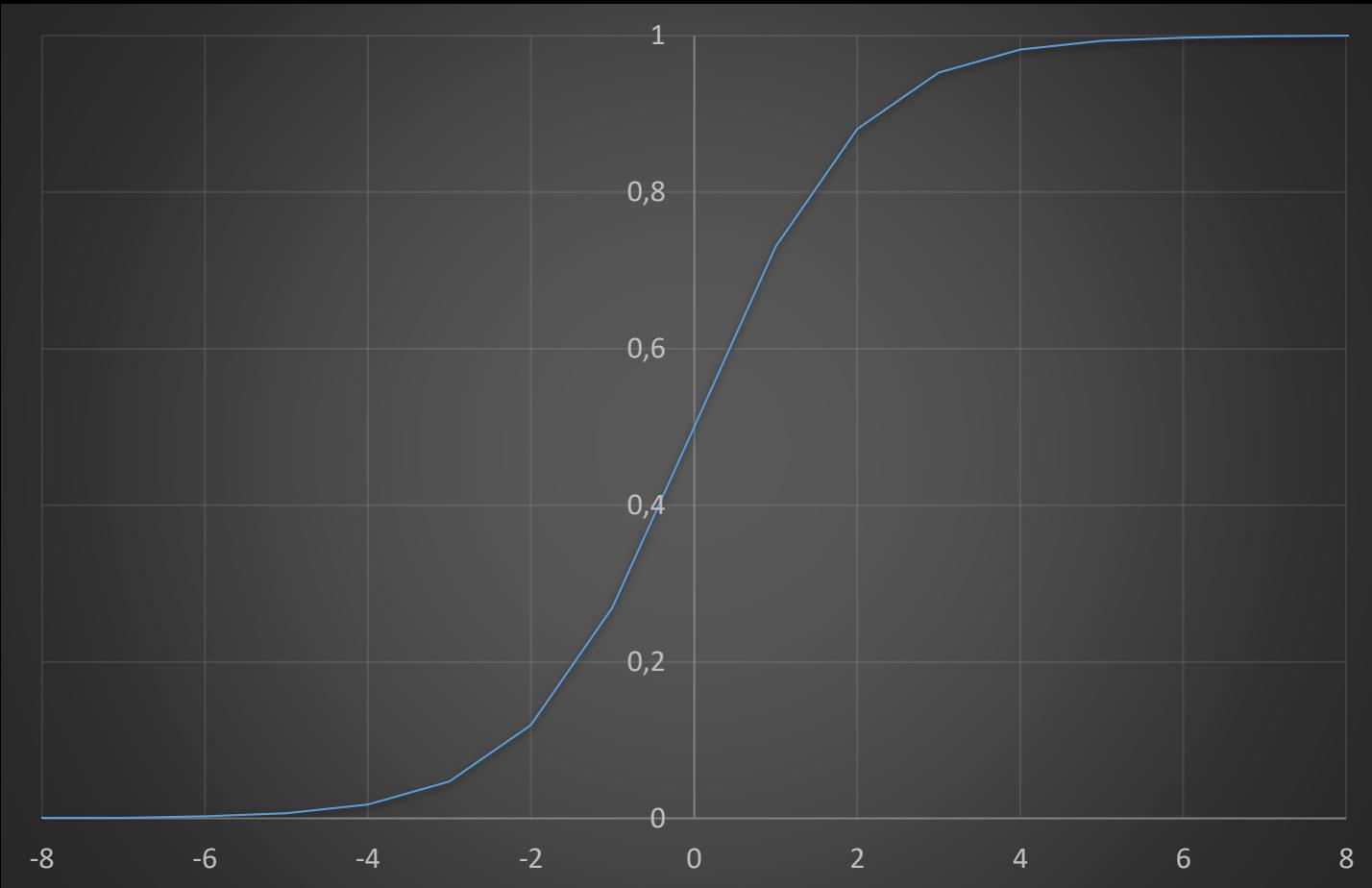
$$f(x) = \begin{cases} 1 & \text{if } x \geq \theta \\ -1 & \text{if } x < \theta \end{cases}$$

Fungsi Aktivasi

- 4. Fungsi sigmoid biner/*logistic sigmoid*:

$$f(x) = \frac{1}{1 + e^{-x}}$$

- Output: 0,0–1,0



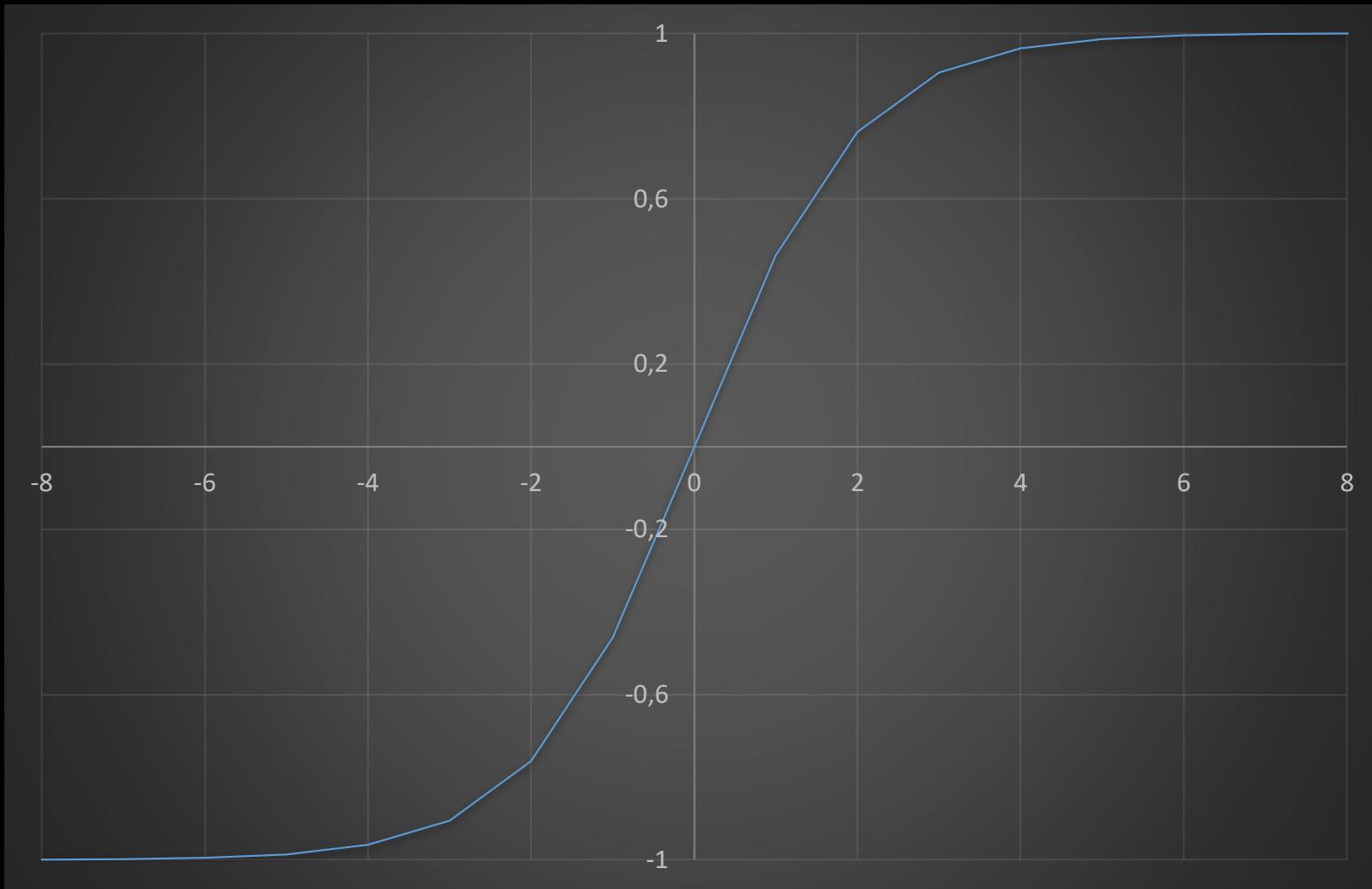
Fungsi Sigmoid Biner

Fungsi Aktivasi

5. Sigmoid bipolar:

$$f(x) = \frac{2}{1 + e^{-x}} - 1$$

- Output: $-1 \dots 1$



Fungsi Sigmoid Bipolar

The background of the image is a dark, almost black, space. Overlaid on this are numerous thin, blue, glowing lines that represent the soma and dendrites of neurons. These lines are interconnected by small, glowing blue dots, representing synapses. The overall effect is a complex, organic, and glowing network.

McCulloch-Pitts Neuron

BULLETIN OF
MATHEMATICAL BIOPHYSICS
VOLUME 5, 1943

A LOGICAL CALCULUS OF THE
IDEAS IMMANENT IN NERVOUS ACTIVITY

WARREN S. McCULLOCH AND WALTER PITTS

FROM THE UNIVERSITY OF ILLINOIS, COLLEGE OF MEDICINE,
DEPARTMENT OF PSYCHIATRY AT THE ILLINOIS NEUROPSYCHIATRIC INSTITUTE,
AND THE UNIVERSITY OF CHICAGO

Because of the "all-or-none" character of nervous activity, neural events and the relations among them can be treated by means of propositional logic. It is found that the behavior of every net can be described in these terms, with the addition of more complicated logical means for nets containing circles; and that for any logical expression satisfying certain conditions, one can find a net behaving in the fashion it describes. It is shown that many particular choices among possible neurophysiological assumptions are equivalent, in the sense that for every net behaving under one assumption, there exists another net which behaves under the other and gives the same results, although perhaps not in the same time. Various applications of the calculus are discussed.

McCulloch-Pitts Neuron

- Dirancang oleh Warren **McCulloch** (1898–1969) dan Walter **Pitts** (1923–1969)
- Diperkirakan merupakan arsitektur jaringan saraf yang **pertama** kali dibuat (1943)
- Walaupun sederhana, namun telah memperkenalkan beberapa aspek penting dari jaringan saraf

McCulloch-Pitts Neuron

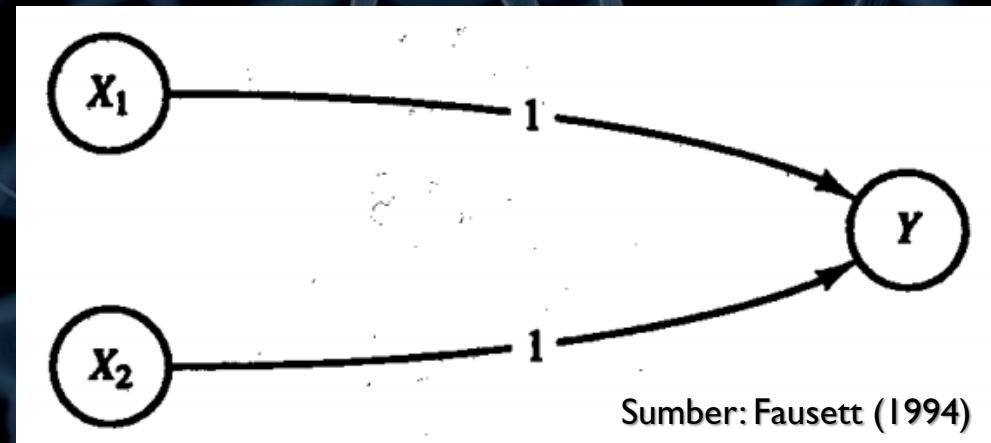
- Menggunakan fungsi aktivasi biner (0/1)
- Setiap neuron memiliki suatu nilai *threshold* (θ) yang ditentukan
$$f(x) = \begin{cases} 1 & \text{if } x \geq \theta \\ 0 & \text{if } x < \theta \end{cases}$$
- Nilai-nilai bobot juga ditentukan

Logika AND

- Berapakah nilai *threshold* (θ) yang harus digunakan untuk membentuk logika AND?

$$f(x) = \begin{cases} 1 & \text{if } x \geq \theta \\ 0 & \text{if } x < \theta \end{cases}$$

x_1	x_2	y
1	1	1
1	0	0
0	1	0
0	0	0



Sumber: Fausett (1994)

Implementasi

```
import numpy as np

# Fungsi step biner
def binstep(x, th=0):
    return 1 if x >= th else 0
```

Implementasi

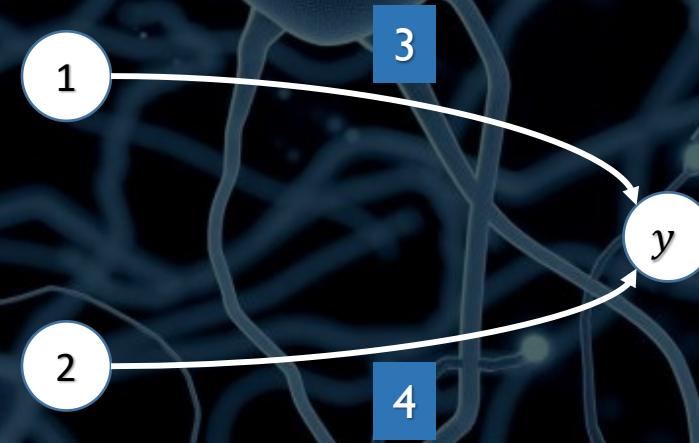
- Perkalian dan penjumlahan antara neuron dan bobot adalah operasi *dot product*
- Dapat dilakukan menggunakan **np.dot**:

```
n = [1, 2]
```

```
w = [3, 4]
```

```
y = np.dot(n, w)
```

```
print(y) # 11
```



Implementasi

```
# Logika AND dengan nilai bobot [1, 1]

def AND(x):
    y_in = np.dot(x, [1, 1])

    return binstep(y_in, 2)

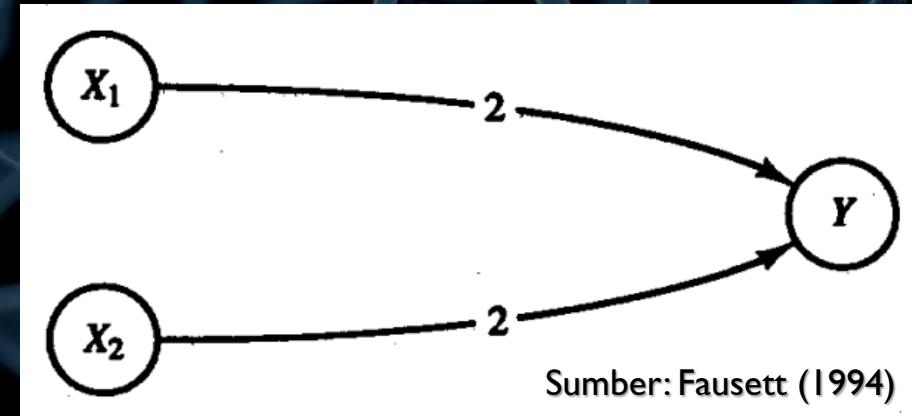
print(AND([1, 1]))
```

Logika OR

- Berapakah nilai *threshold* (θ) yang harus digunakan untuk membentuk logika OR?

$$f(x) = \begin{cases} 1 & \text{if } x \geq \theta \\ 0 & \text{if } x < \theta \end{cases}$$

x_1	x_2	y
1	1	1
1	0	1
0	1	1
0	0	0



Sumber: Fausett (1994)

Implementasi

```
# Logika OR dengan nilai bobot [2, 2]

def OR(x):
    y_in = np.dot(x, [2, 2])

    return binstep(y_in, 2)

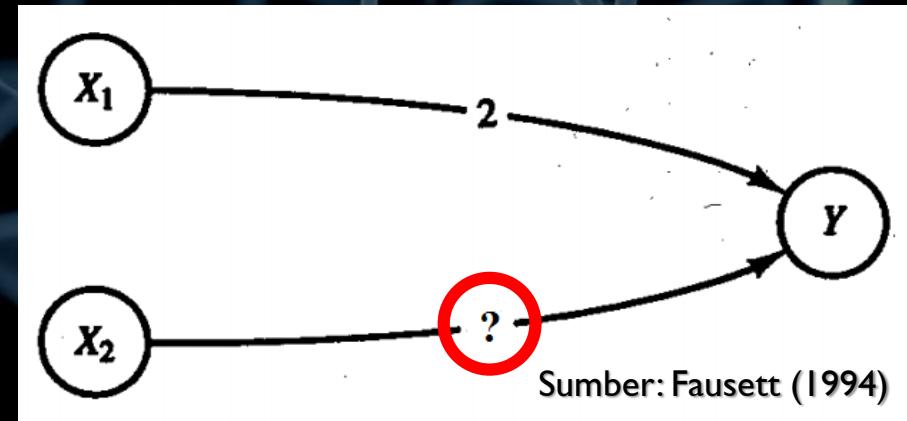
print(OR([1, 1]))
```

Logika AND NOT

- Jika nilai *threshold* ditetapkan sebesar 2, berapakah nilai w_2 yang harus digunakan untuk membentuk logika AND NOT?

$$f(x) = \begin{cases} 1 & \text{if } x \geq \theta \\ 0 & \text{if } x < \theta \end{cases}$$

x_1	x_2	y
1	1	0
1	0	1
0	1	0
0	0	0



Sumber: Fausett (1994)

Implementasi

```
# Logika AND NOT dengan nilai bobot [2, -1]

def ANDNOT(x):
    y_in = np.dot(x, [2, -1])

    return binstep(y_in, 2)

print(ANDNOT([1, 0]))
```

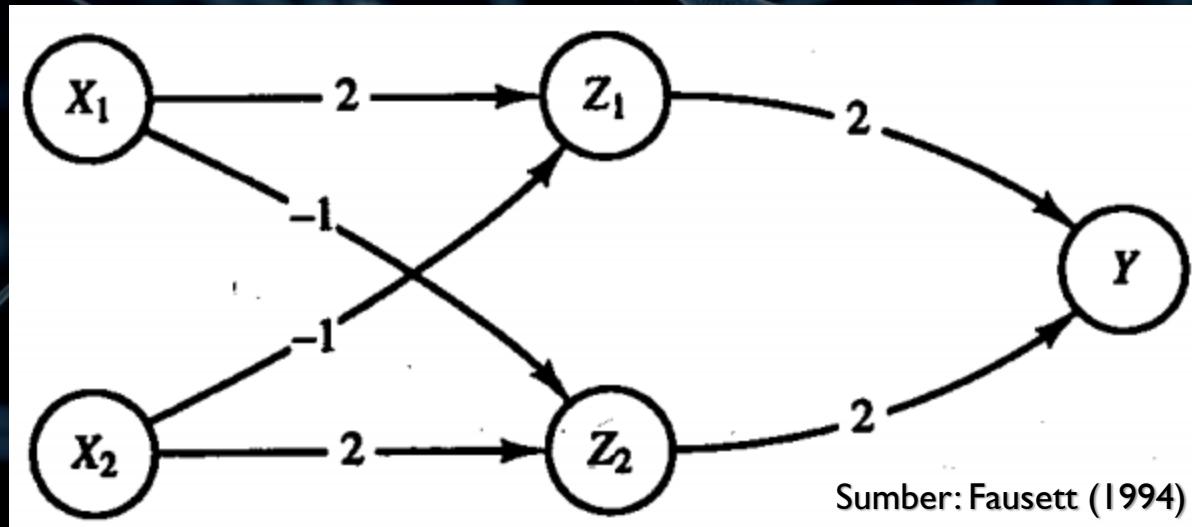
Logika XOR

- Dibentuk dari gabungan dua logika AND NOT

x_1	x_2	$x_1 \text{ AND } \text{NOT } x_2$ (y_1)	$x_2 \text{ AND } \text{NOT } x_1$ (y_2)	$y_1 \text{ OR } y_2$
1	1	0	0	0
1	0	1	0	1
0	1	0	1	1
0	0	0	0	0

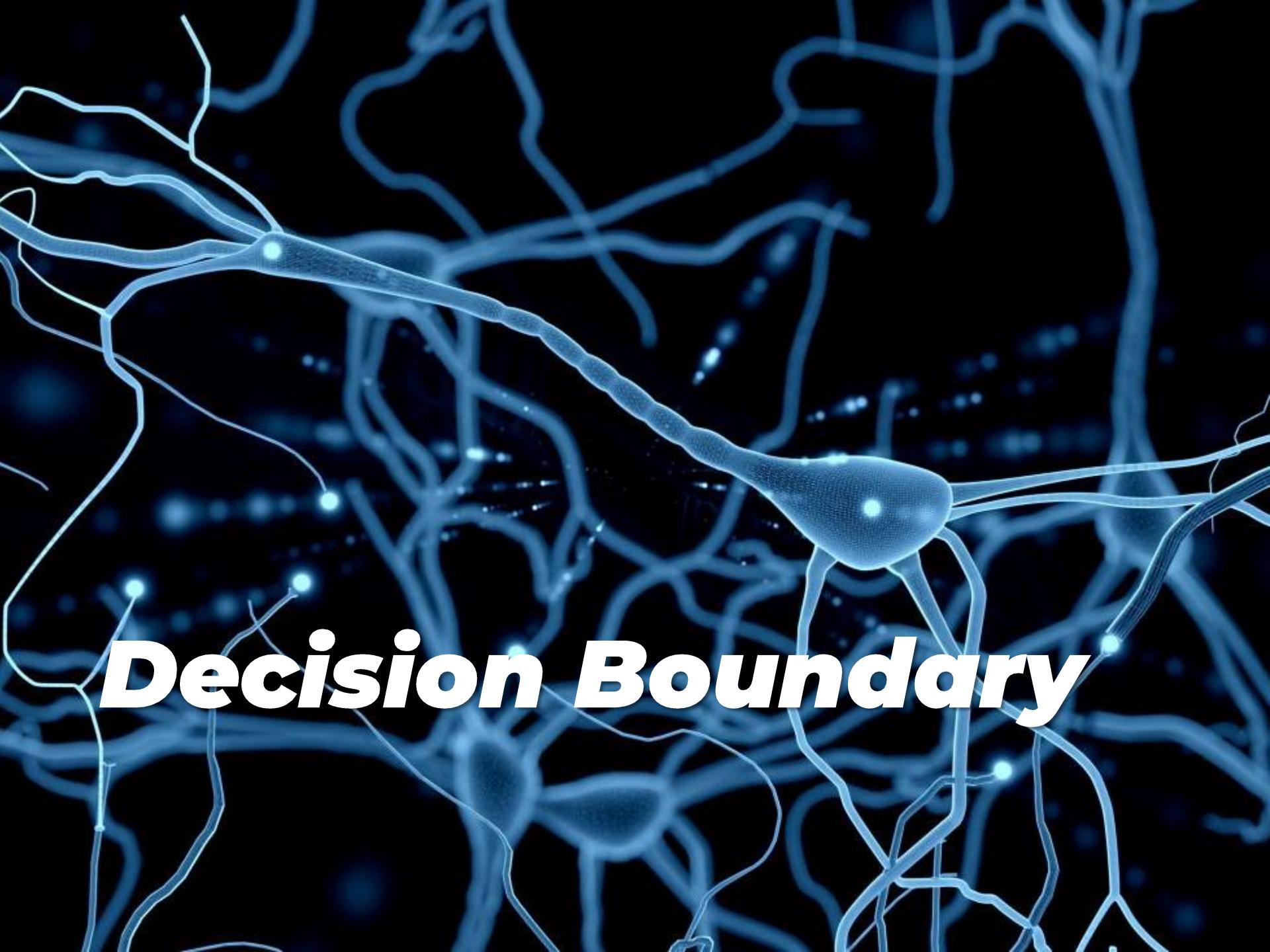
Logika XOR

- *Threshold (θ) bernilai 2*



Implementasi

```
# Logika XOR yang merupakan gabungan antara  
# logika AND NOT dan OR  
  
def XOR(x):  
    # x[::-1] membalik urutan array x  
    z = [ANDNOT(x), ANDNOT(x[::-1])]  
  
    return OR(z)  
  
print(XOR([1, 1]))
```

The background of the image is a dark, almost black, space. Overlaid on it is a complex, glowing blue network of lines and dots, resembling a neural network or a complex circuit. The lines represent the connections between neurons, and the glowing dots represent the neurons themselves. Some dots are larger and more intense than others, suggesting they are more active or central. The overall effect is one of a living, breathing system.

Decision Boundary

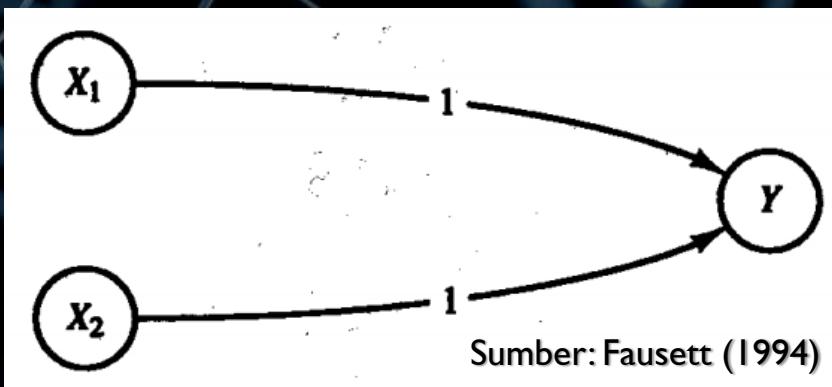
Decision Boundary

- Logika AND dapat dipandang sebagai sebuah permasalahan klasifikasi
- x_1 dan x_2 sebagai atribut
- y sebagai kelas

x_1	x_2	y
1	1	1
1	0	0
0	1	0
0	0	0

Decision Boundary

- $net = x_1w_1 + x_2w_2$
- Jika $net \geq 2$, maka data tersebut masuk ke kelas positif (+)
- Jika $net < 2$, maka data tersebut masuk ke kelas negatif (-)

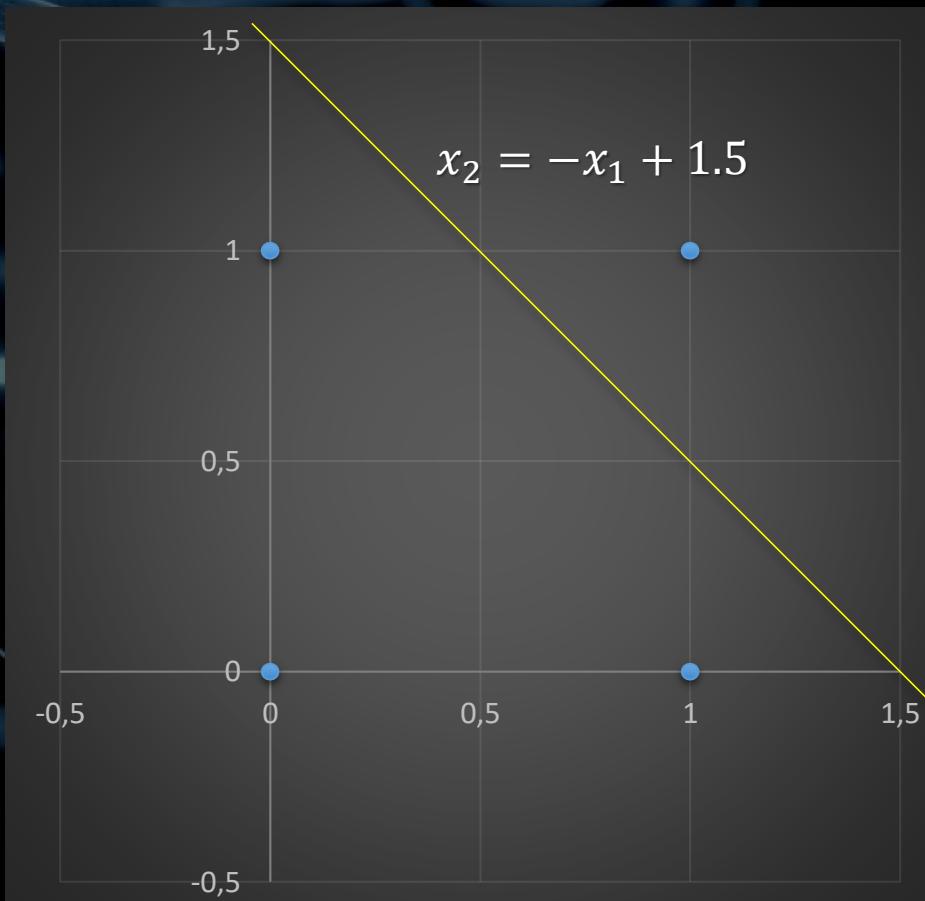


x_1	x_2	net	Kelas
1	1	2	+
1	0	1	-
0	1	1	-
0	0	0	-

Decision Boundary

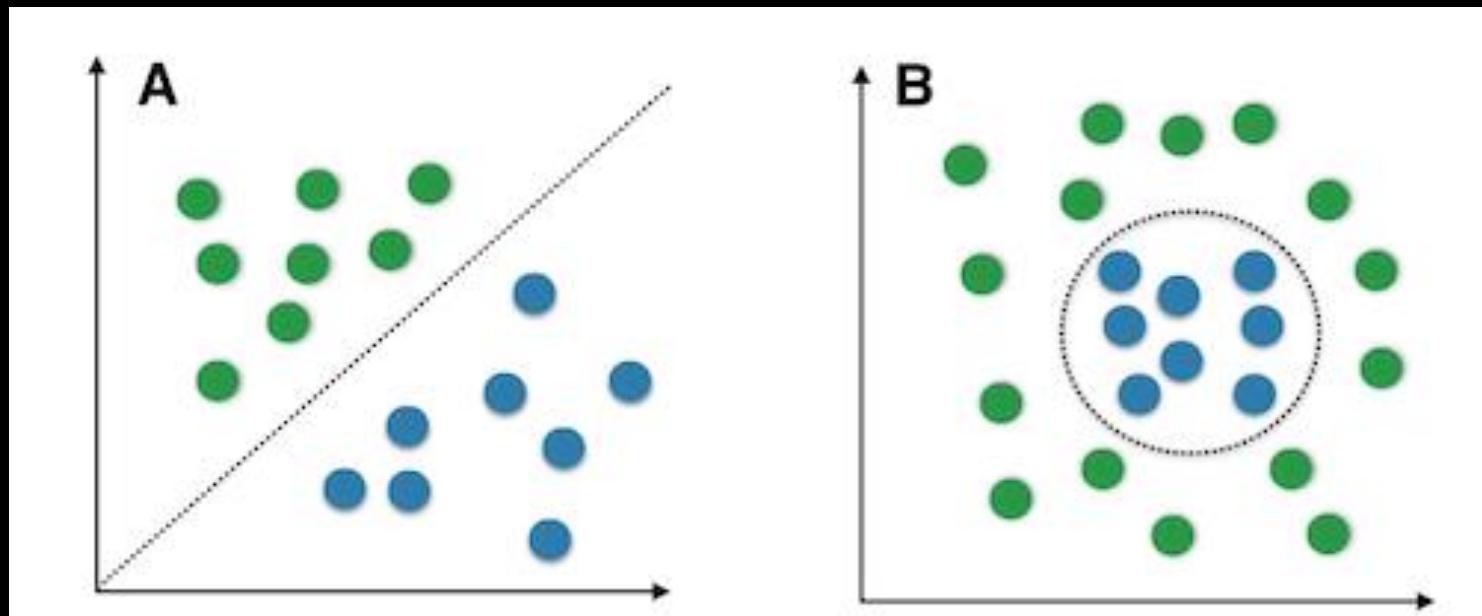
- Nilai tengah 1,5 adalah nilai pemisah antara kedua kelas tersebut
- $net = x_1w_1 + x_2w_2$
- $1.5 = x_1 + x_2$
- $x_2 = -x_1 + 1.5$
- Garis pemisah itu disebut dengan *decision boundary*

Decision Boundary



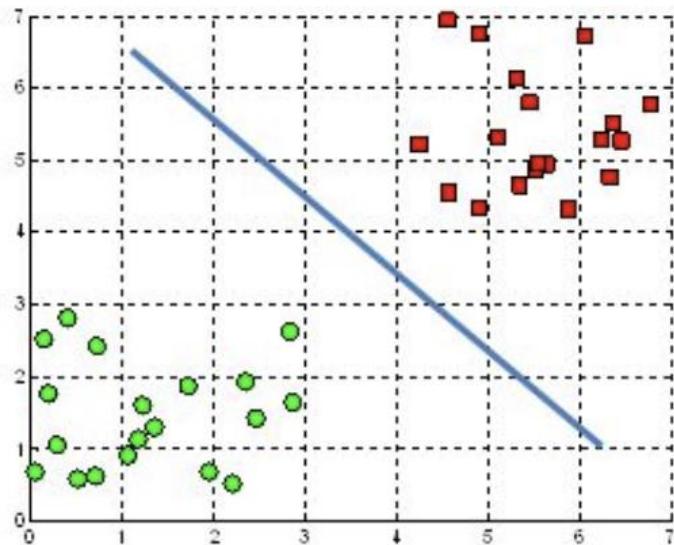
Decision Boundary

- Karena kedua kelas tersebut dapat dipisahkan oleh sebuah garis lurus (linier), maka kedua kelas itu disebut *linearly separable*

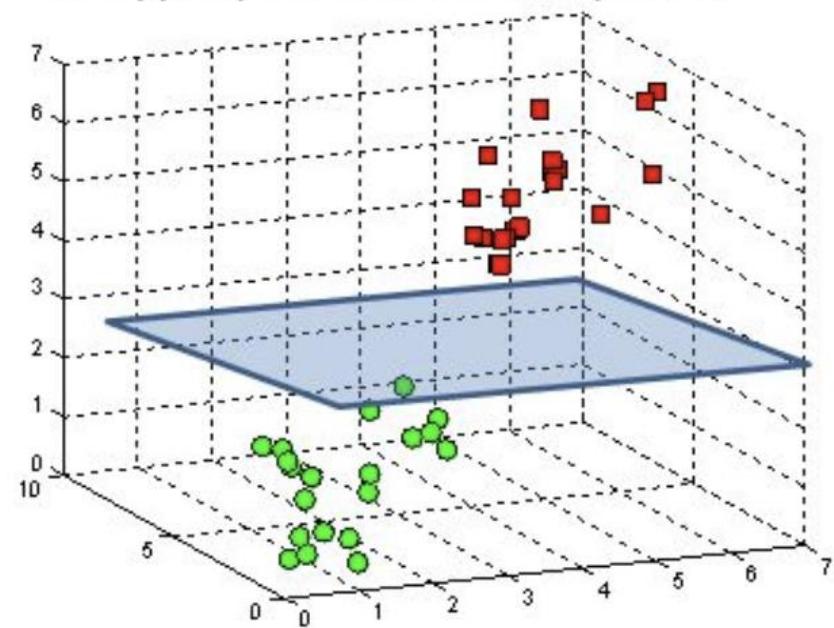


Linearly separable vs. non-linearly separable

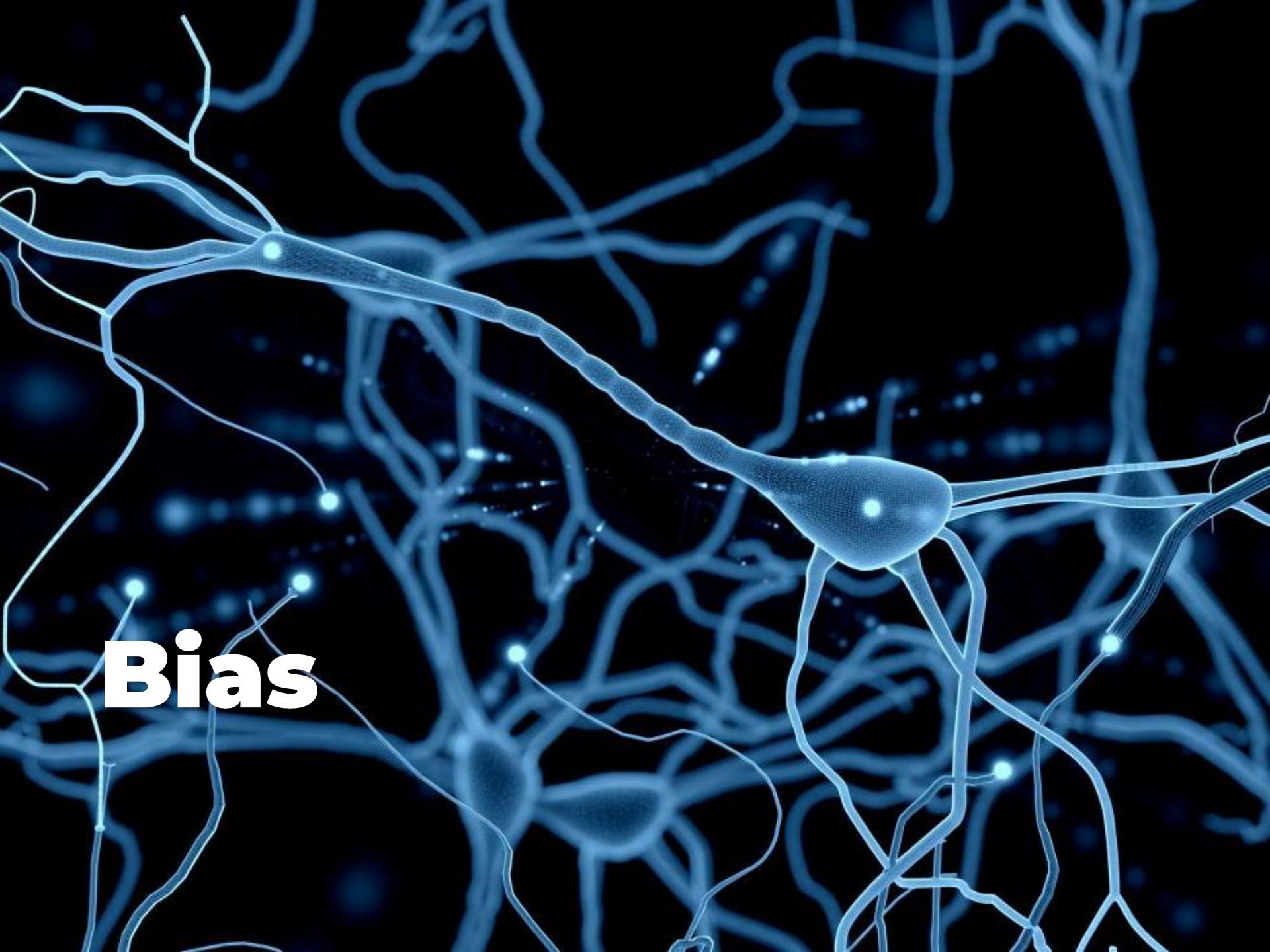
A hyperplane in \mathbb{R}^2 is a line



A hyperplane in \mathbb{R}^3 is a plane



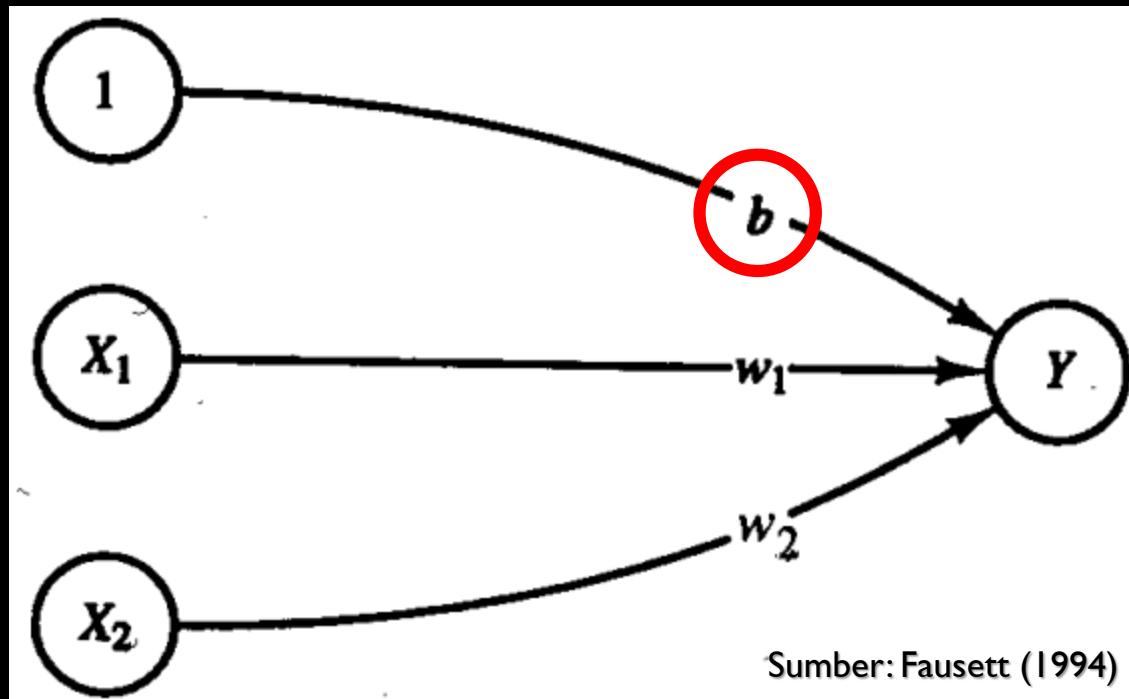
Linear decision boundaries

A dense network of glowing blue neurons against a black background. The neurons are depicted as translucent blue spheres representing cell bodies, with numerous thin, branching lines representing axons and dendrites extending in various directions. Some neurons are brightly lit, while others are more dimly visible in the background.

Bias

Bias

- Bias adalah bobot dari sebuah neuron yang nilai aktivasinya selalu bernilai 1
- Nilai biasnya sendiri dapat berubah-ubah



Bias

- Jika

$$net = \sum_i x_i w_i$$

maka kemudian dapat diaplikasikan fungsi aktivasi
step:

$$f(net) = \begin{cases} 1, & \text{if } net \geq \theta \\ -1, & \text{if } net < \theta \end{cases}$$

- *Decision boundary* antara kedua kelas adalah:

$$x_1 w_1 + x_2 w_2 = \theta$$

Bias

- Dengan bias, nilai *net* dihitung sebagai:

$$net = b + \sum_i x_i w_i$$

- Maka, fungsi aktivasinya menjadi:

$$f(net) = \begin{cases} 1, & \text{if } net \geq 0 \\ -1, & \text{if } net < 0 \end{cases}$$

- *Decision boundary* antara kedua kelas adalah:

$$b + x_1 w_1 + x_2 w_2 = 0$$

Bias

- Penggunaan bias dan nilai *threshold* nol memberikan **hasil yang sama** dengan kondisi jika bias tidak digunakan dan *threshold* bernilai selain nol
- Proses *training* akan mengarahkan jaringan untuk memberikan hasil yang benar

Logika AND Tanpa Bias

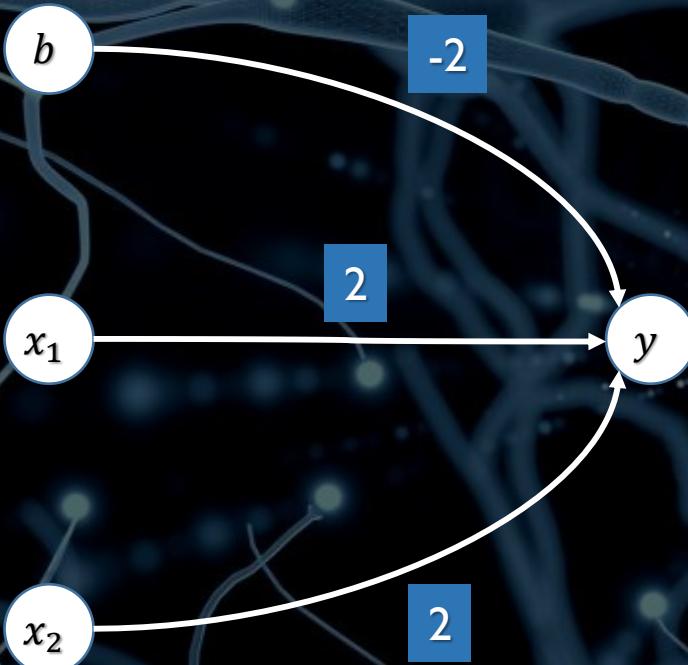


x_1	x_2	y
1	1	1
1	0	0
0	1	0
0	0	0

$$f(\text{net}) = \begin{cases} 1, & \text{if } \text{net} \geq 2 \\ 0, & \text{if } \text{net} < 2 \end{cases}$$

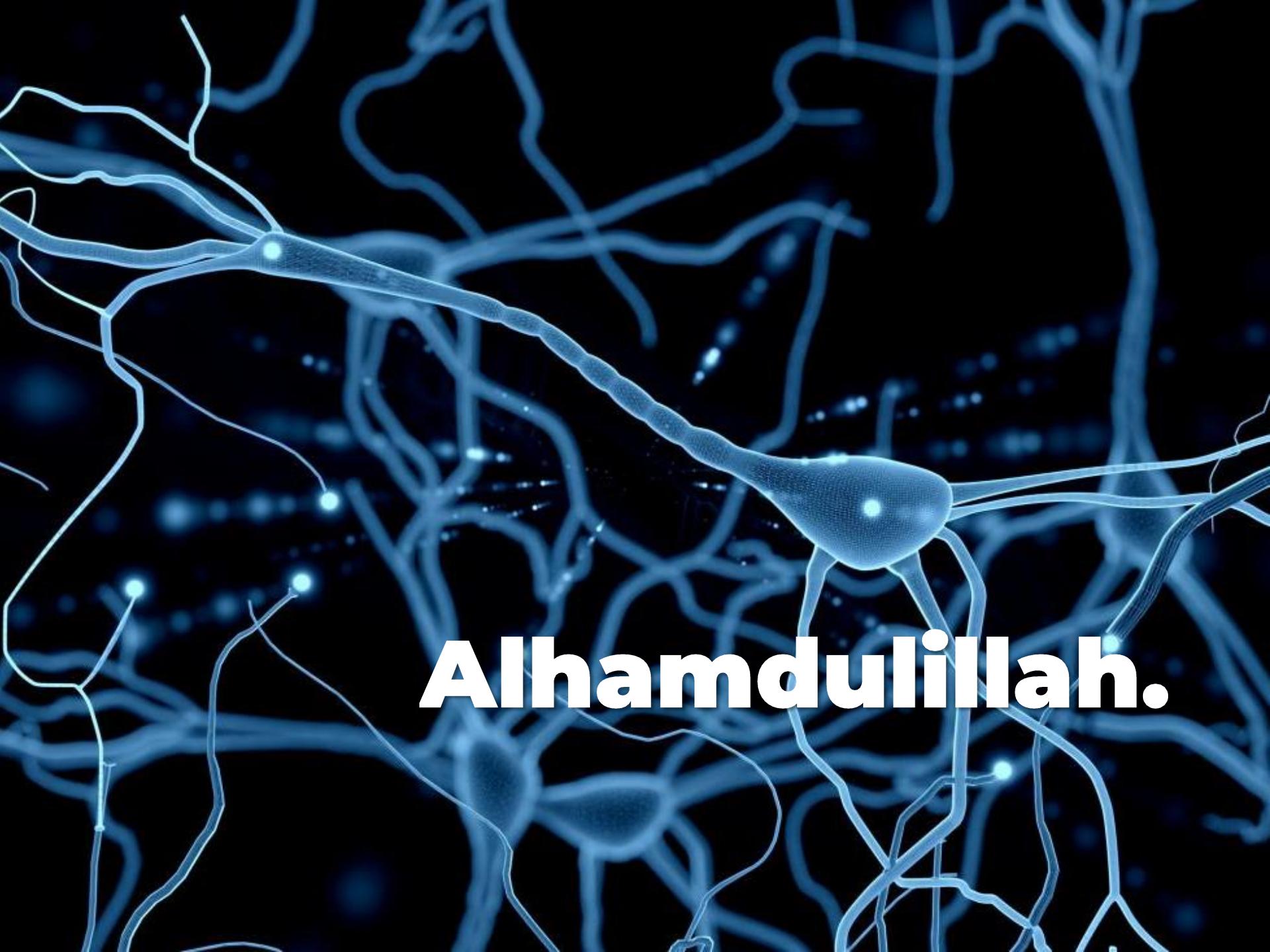
Logika AND dengan Bias

(Hasil pelatihan dengan Hebb rule)



x_1	x_2	y
1	1	1
1	-1	-1
-1	1	-1
-1	-1	-1

$$f(\text{net}) = \begin{cases} 1, & \text{if } \text{net} \geq 0 \\ -1, & \text{if } \text{net} < 0 \end{cases}$$

A dense network of glowing blue neurons against a black background.

Alhamdulillah.