



JARINGAN SARAF TIRUAN

ADALINE (Adaptive Linear Neuron)



Pendahuluan

- Model ADALINE (Adaptive Linear Neuron) ditemukan oleh Widrow & Hoff (1960).
- Arsitekturnya mirip dengan perceptron. Beberapa masukan (dan sebuah bias) dihubungkan langsung dengan sebuah neuron keluaran. Perbedaan dengan perceptron adalah dalam hal cara modifikasi bobotnya.
- Bobot dimodifikasi dengan aturan delta (sering juga disebut least mean square).



- Selama pelatihan, fungsi aktivasi yang dipakai adalah fungsi identitas

$$\text{net} = \sum_i x_i w_i + b$$

$$y = f(\text{net}) = \text{net} = \sum_i x_i w_i + b$$

- Error adalah Kuadrat selisih antara target (t) dan keluaran jaringan (f(net)), bobot dimodifikasi sedemikian hingga errornya minimum.

$$E = (t - f(\text{net}))^2 = \left(t - \left(\sum_i x_i w_i + b \right) \right)^2$$

- Perubahan bobot adalah : $\Delta w_i = \alpha(t - y)x_i$ dengan α merupakan bilangan positif kecil (umumnya diambil = 0,1)



Algoritma pelatihan ADALINE adalah sebagai berikut :

1. Inisialisasi semua bobot dan bias (umumnya $w_i = b = 0$)

Tentukan laju pemahaman ($=\alpha$). Untuk penyederhanaan, biasanya α diberi nilai kecil ($= 0.1$)

Tentukan toleransi kesalahan yang diijinkan

2. Selama $\max_i \Delta w_i > \text{batas toleransi}$, lakukan :

- a. Set aktivasi unit masukan $x_i = s_i$ ($i = 1, \dots, n$)

- b. Hitung respon unit keluaran : $\text{net} = \sum_i x_i w_i + b$

$$y = f(\text{net}) = \text{net}$$

- c. Perbaiki bobot pola yang mengandung kesalahan ($y \neq t$) menurut persamaan :

$$w_i(\text{baru}) = w_i(\text{lama}) + \alpha (t - y) x_i$$

$$b(\text{baru}) = b(\text{lama}) + \alpha (t - y)$$



- Setelah proses pelatihan selesai, ADALINE dapat dipakai untuk pengenalan pola.
- Untuk itu, umumnya dipakai fungsi threshold bipolar (meskipun tidak menutup kemungkinan digunakan bentuk lainnya). Caranya adalah sebagai berikut :

1. Inisialisasi semua bobot dan bias dengan bobot dan bias hasil pelatihan
2. Untuk setiap input masukan bipolar x , lakukan :
 - a. Set aktivasi unit masukan $x_i = s_i$ ($i = 1, \dots, n$)
 - b. Hitung net vektor keluaran :

$$\text{net} = \sum_i x_i w_i + b$$

- c. Kenakan fungsi aktivasi :

$$y = \begin{cases} 1 & \text{jika } \text{net} \geq 0 \\ -1 & \text{jika } \text{net} < 0 \end{cases}$$



Contoh :

Gunakan model ADALINE untuk mengenali pola fungsi logika “dan” dengan masukan dan target bipolar :

Masukan		Target
x_1	x_2	t
1	1	1
1	-1	-1
-1	1	-1
-1	-1	-1

Gunakan batas toleransi = 0.05 dan $\alpha = 0.1$



Penyelesaian

Dengan $\alpha = 0.1$, maka perubahan bobotnya $\Delta w_i = \alpha(t - f(\text{nett}))x_i = 0.1(t - y)x_i$. Iterasi untuk epoch-1 tampak pada table:

Masukan		y =			Perubahan Bobot			Bobot Baru		
(x ₁ x ₂ 1)	t	net	f (net)	t - y	(Δw ₁ Δw ₂ Δb)	(w ₁ w ₂ b)				
Inisialisasi						(0 0 0)				
(1 1 1)	1	0	0	1	(0.1 0.1 0.1)	(0.1 0.1 0.1)				
(1 -1 1)	-1	0.1	0.1	-1.1	(-0.11 0.11 -0.11)	(-0.01 0.21 -0.01)				
(-1 1 1)	-1	0.21	0.21	-1.21	(0.12 -0.12 -0.12)	(0.11 0.09 -0.13)				
(-1 -1 1)	-1	-0.33	-0.33	-0.67	(0.07 0.07 -0.07)	(0.18 0.16 -0.2)				



Lanjutan....

Maksimum $\Delta w_i = 0.07 > \text{toleransi}$, maka iterasi dilanjutkan untuk epoch kedua, yang tampak pada table

Masukan			y =		Perubahan Bobot			Bobot Baru		
(x_1 x_2 1)	t	net	f (net)	t - y	(Δw_1 Δw_2 Δb)	(w ₁ w ₂ b)				
Inisialisasi							(0.18	0.16	-0.2)	
(1 1 1)	1	0.14	0.14	0.86	(0.09 0.09 0.09)	(0.26	0.24	-0.11)		
(1 -1 1)	-1	-0.09	-0.09	-0.91	(-0.09 0.09 -0.09)	(0.17	0.33	-0.2)		
(-1 1 1)	-1	-0.04	-0.04	-0.96	(0.1 -0.1 -0.1)	(0.27	0.24	-0.3)		
(-1 -1 1)	-1	-0.8	-0.8	-0.2	(0.02 0.02 -0.02)	(0.29	0.26	-0.32)		

Maksimum $\Delta w_i = 0.02 < \text{toleransi}$, maka iterasi dihentikan dan bobot terakhir yang diperoleh ($w_1 = 0.29$, $w_2 = 0.26$ dan $b = -0.32$) merupakan bobot yang digunakan dalam pengenalan polanya



Lanjutan...

Pengenalan pola fungsi “dan” menggunakan bobot hasil pelatihan. Perhatikan bahwa fungsi aktivasi yang dipakai berbeda dengan fungsi aktivasi pada pelatihan.

Dalam pengenalan pola, fungsi aktivasinya adalah:

$$y = \begin{cases} 1 & \text{jika } net \geq 0 \\ -1 & \text{jika } net < 0 \end{cases}$$

Masukan		net	y
x ₁	x ₂		
1	1	0.23	1
1	-1	-0.29	-1
-1	1	-0.35	-1
-1	-1	-0.87	-1

Tampak bahwa keluaran jaringan tepat sama dengan targetnya. Disimpulkan bahwa pola dapat dikenali dengan sempurna menggunakan bobot hasil pelatihan



Contoh :

Menggunakan $\alpha = 0.2$,

Masukan		y =			Perubahan Bobot			Bobot Baru		
(x ₁ x ₂ 1)	t	net	f (net)	t - y	(Δw ₁ Δw ₂ Δb)	(w ₁ w ₂ b)				
Inisialisasi						(0 0 0)				
(1 1 1)	1	0	0	1	(0.2 0.2 0.2)	(0.2	0.2	0.2)		
(1 -1 1)	-1	0.2	0.2	-1.2	(-0.24 0.24 -0.24)	(-0.04	0.44	-0.04)		
(-1 1 1)	-1	0.44	0.44	-1.44	(0.29 -0.29 -0.29)	(0.25	0.15	-0.33)		
(-1 -1 1)	-1	-0.73	-0.73	-0.27	(0.05 0.05 -0.05)	(0.3	0.21	-0.38)		



- Maksimum $\Delta w_i = 0.05$ = toleransi, maka iterasi dihentikan dan bobot terakhir yang diperoleh ($w_1 = 0.3$, $w_2 = 0.21$ dan $b = -0.38$) merupakan bobot yang digunakan dalam pengenalan polanya. Dengan cara seperti perhitungan tabel 6.4, dapat dicek bahwa bobot yang diperoleh akan mengenali semua pola dengan benar
- Dari contoh tampak bahwa penggunaan α yang lebih besar akan menyebabkan iterasi menjadi lebih cepat. Akan tetapi penggunaan α yang terlalu besar akan menyebabkan iterasi melompat terlalu jauh sehingga melewati bobot optimalnya.



Soal Latihan :

- Gunakan ADALINE untuk mengenali fungsi logika “dan” dengan 3 buah masukan ($\eta = 0.5$ dan toleransi = 0.1).