KONSEP MATEMATIKA DI BALIK JARINGAN SARAF TIRUAN SEBAGAI FONDASI KECERDASAN BUATAN

*E. Khatizah

Departemen Matematika, FMIPA,
Institut Pertanian Bogor, Jl. Meranti, Kampus IPB Dramaga Bogor.
elis_khatizah@apps.ipb.ac.id.***corresponding author**

Abstrak

Kecerdasan buatan (AI) telah menjadi salah satu teknologi yang berpengaruh di berbagai sektor, mulai dari kesehatan hingga industri otomotif. Di balik kemajuan ini, terdapat dasar matematika yang memainkan peran penting, khususnya dalam proses optimasi dan pembelajaran mesin, dua elemen utama pendukung kinerja AI. Artikel ini bertujuan untuk mengulas konsep dasar matematika, khususnya kalkulus turunan, yang berperan dalam pembelajaran jaringan saraf tiruan sebagai bagian dari konstruksi model AI. Dengan penjelasan teori dan contoh praktis, artikel ini memaparkan kontribusi matematika dalam mendasari dan membentuk model AI. Melalui pemahaman ini, diharapkan pembaca tidak hanya melihat matematika sebagai teori semata, tetapi juga sebagai alat esensial untuk membangun teknologi masa depan.

Kata kunci: *gradient descent*, jaringan saraf tiruan, kalkulus turunan, kecerdasan buatan

1 Pendahuluan

Kecerdasan buatan (AI) telah menjadi pendorong utama di balik inovasi teknologi dalam berbagai aspek kehidupan, seperti layanan kesehatan, transportasi, dan hiburan. Ditinjau dari pengertiannya, AI mengacu pada program komputer yang meniru kecerdasan manusia [2]. Salah satu cabang penting dalam AI adalah pembelajaran mesin (*machine learning*), yang memungkinkan sistem untuk belajar dari pengalaman untuk meningkatkan performanya. Di dalamnya, terdapat teknik pembelajaran mendalam (*deep learning*) yang didasarkan pada jaringan saraf tiruan (*artificial neural network*) [5]. Dari sini dapat kita katakan, jaringan saraf tiruan merupakan salah satu pilar pembentuk fondasi AI. Selanjutnya, teknologi pengembangan AI memungkinkan komputer mempelajari pola, membuat prediksi, dan mengambil keputusan berdasarkan data yang tersedia. Di balik kemudahan yang dirasakan pengguna, sebetulnya terdapat proses matematis yang kompleks yang menjadi inti dari kinerja AI. Pemahaman terhadap proses matematis ini sangat penting, terutama bagi generasi yang akan membangun inovasi AI di masa depan.

Namun, banyak kalangan, termasuk mahasiswa dari berbagai latar belakang pendidikan, sering kali tidak menyadari bahwa di balik kecanggihan AI terdapat dasar matematika yang kuat. Salah satu cabang matematika yang mendukung perkembangan ini adalah kalkulus, yang menjadi landasan dalam proses pembelajaran mesin dan optimasi model AI [3]. Dalam konteks ini, kalkulus menjadi tulang punggung proses

2020 Mathematics Subject Classification: 00A69, 00A06.
Diajukan:13/12/2024, diterima: 31/12/2024. DOI: https://doi.org/10.29244/milang.20.2.145-156

optimasi melalui konsep turunan, yang memungkinkan algoritme AI melakukan penyesuaian bobot dan parameter model untuk meningkatkan akurasi prediksi dari waktu ke waktu. Kita tidak bisa memungkiri bahwa aplikasi matematika dalam kehidupan sehari-hari sering kali tidak terlihat secara langsung sehingga berdampak pada rendahnya apresiasi mahasiswa terhadap pembelajaran matematika di perguruan tinggi [1]. Padahal, pengetahuan matematika semisal kalkulus dasar tidak hanya relevan bagi mahasiswa jurusan matematika, tetapi juga bagi semua mahasiswa yang ingin memahami bagaimana teknologi canggih seperti AI berkembang dari prinsip-prinsip dasar matematika.

Berangkat dari hal tersebut, artikel ini bertujuan untuk membuka wawasan mahasiswa secara umum tentang pentingnya konsep matematika sederhana dalam AI, sekaligus menjadi motivasi mahasiswa matematika untuk terus tekun mendalami keilmuan mereka. Dengan mengambil sebagian konsep kalkulus dasar yang berfokus pada penerapan turunan dalam proses pelatihan jaringan saraf tiruan untuk pembentukan model klasifikasi, artikel ini akan menjelaskan secara sederhana peran konsep matematika dalam membentuk fondasi AI. Tulisan ini juga dilengkapi dengan contoh praktis untuk memudahkan pemahaman konsep. Dengan memahami dasar-dasar kalkulus yang digunakan dalam jaringan saraf tiruan, diharapkan pembaca dapat melihat bahwa matematika merupakan alat esensial yang mendukung pengembangan model AI yang lebih efisien, yang merupakan kunci dari kemajuan teknologi yang kita nikmati hari ini.

2 Konsep Kalkulus Dasar dalam Konstruksi dan Pembelajaran Jaringan Saraf Tiruan

Pada bagian ini, dibahas konsep kalkulus dasar yang berperan penting dalam membentuk inti jaringan saraf tiruan. Jaringan ini dapat diibaratkan sebagai mesin yang melakukan optimasi dengan cara belajar dari data yang diberikan, sehingga menghasilkan model yang meminimumkan kesalahan. Konsep utama yang diulas meliputi turunan, nilai minimum/maksimum fungsi, dan algoritma *gradient descent*.

2.1 Turunan

Turunan adalah konsep dasar dalam kalkulus yang mengukur laju perubahan suatu fungsi terhadap variabel bebasnya. Secara matematis, turunan suatu fungsi pada titik tertentu didefinisikan sebagai limit dari perbandingan perubahan nilai fungsi terhadap perubahan nilai variabel bebas ketika perubahan variabel tersebut mendekati nol. Definisi formalnya dapat dinyatakan sebagai $f'(x) = \lim_{h \to 0} \frac{f(x+h)-f(x)}{h}$. Secara geometris, $f'(a)$ merepresentasikan kemiringan garis singgung pada kurva kurva f dititik $x = a$.

Untuk menghitung turunan suatu fungsi secara praktis tanpa melalui proses limit, dapat digunakan beberapa aturan dasar turunan. Aturan-aturan tersebut adalah sebagai berikut [10]:

$$\frac{d}{dx}(c) = 0,$$
$$\frac{d}{dx}(x^n) = nx^{n-1},$$
$$\frac{d}{dx}(f(x) \pm g(x)) = f'(x) \pm g'(x),$$
$$\frac{d}{dx}\big(f(x)g(x)\big) = \frac{f'(x)\ g(x) - f(x)g'(x)}{[g(x)]^2},$$
$$\frac{d}{dx}(f(g(x))) = f'(g(x))\ g'(x),$$

dengan c adalah konstanta, n adalah bilangan real serta f dan g adalah fungsi yang memiliki turunan di x.

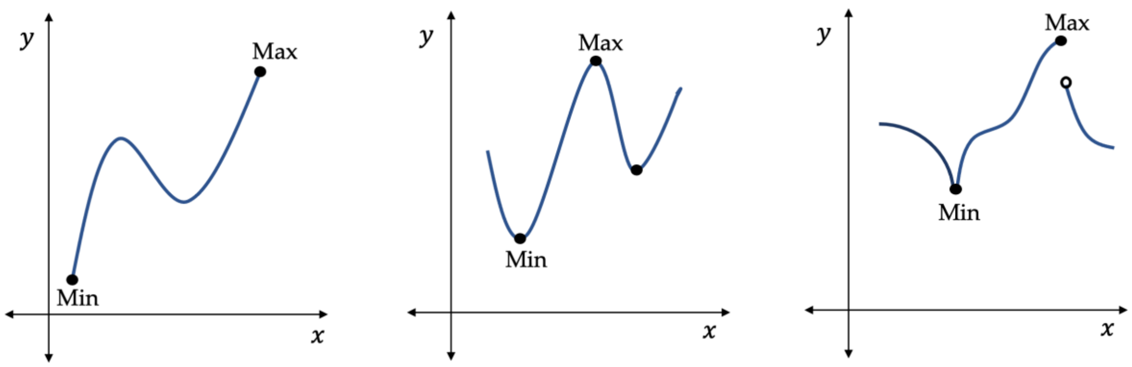
2.2 Nilai Minimum dan Maksimum

Pemahaman tentang titik minimum dan maksimum suatu fungsi sangat penting dalam proses optimasi model AI. Menurut definisi [10], fungsi f memiliki maksimum mutlak di titik c jika $f(c) \geq f(x)$ untuk semua x di daerah asal fungsi f. Nilai $f(c)$ disebut sebagai nilai maksimum fungsi f. Secara serupa, fungsi f memiliki minimum mutlak di titik c jika $f(c) \leq f(x)$ untuk semua x di daerah asal fungsi f. Nilai $f(c)$ disebut sebagai nilai minimum fungsi f.

Teorema nilai maksimum dan minimum menyatakan bahwa jika suatu fungsi f kontinu pada interval tertutup $[a, b]$, maka fungsi tersebut akan mencapai nilai maksimum dan minimum pada interval tersebut. Lebih lanjut, jika $f(c)$ adalah titik ekstrim, maksimum atau minimum, maka c harus merupakan titik kritis [11], yang dapat berupa salah satu dari berikut:

1. titik ujung interval,
2. titik stasioner ($f'(c) = 0$), atau
3. titik singular ($f'(c)$ tidak terdefinisi).

Kondisi ini mempermudah proses pencarian titik di lokasi fungsi mencapai nilai maksimum atau minimum, sebagaimana diilustrasikan pada Gambar 1.



Gambar 1. Ilustrasi fungsi mencapai maksimum dan minimum di (a) titik ujung, (b) titik stasioner, dan (c) titik singular.

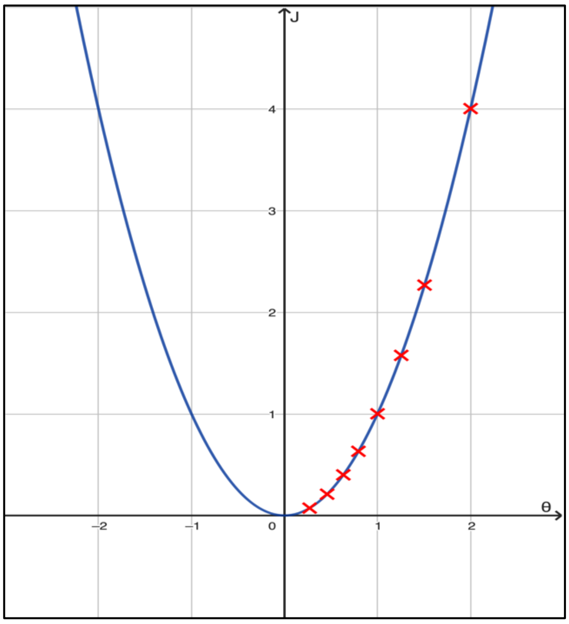
2.3 Algoritma *Gradient Descent*

Pembahasan turunan tidak terlepas dari konsep gradien, yang merupakan turunan dari fungsi terhadap variabel tertentu dan menunjukkan kemiringan atau arah perubahan fungsi. Dalam hal ini, jika kita memiliki fungsi J yang bergantung pada variable θ, maka gradien positif ($\frac{\partial J}{\partial \theta} > 0$) menunjukkan bahwa fungsi naik (menanjak). Sebaliknya, jika gradien negatif ($\frac{\partial J}{\partial \theta} < 0$), berarti fungsi menurun [11]. Konsep gradien digunakan oleh algoritma *gradient descent*, yang diterapkan dalam *machine learning* dan *deep learning* untuk meminimumkan fungsi biaya. Algoritma ini bekerja dengan memperbarui parameter model dalam arah yang berlawanan dengan gradien fungsi objektif (gradien negatif) [9].

Secara lebih rinci, misalkan kita memiliki fungsi biaya $J(\theta)$ yang mengukur seberapa baik model memprediksi data. Untuk meminimumkan fungsi biaya ini, kita menghitung gradien atau turunan dari fungsi biaya terhadap parameter model, yaitu $\frac{\partial J(\theta)}{\partial \theta}$. Selanjutnya, pada iterasi ke$-k$, parameter diperbarui dengan langkah kecil menuju arah gradien negatif menggunakan rumus:

$$\theta^{k+1} = \theta^k - \alpha \frac{\partial J}{\partial \theta}\bigg|_{\theta=\theta^k} \tag{1}$$

dengan α adalah kecepatan langkah (*learning rate*). Proses ini diulang sesuai dengan kriteria penghentian iterasi, salah satunya adalah ketika perubahan nilai fungsi biaya cukup kecil. Gambar 2 mengilustrasikan fungsi J yang berbentuk parabola beserta pembaruan parameter θ yang dilakukan berdasarkan Persamaan (1) hingga mencapai nilai parameter yang meminimumkan fungsi J.



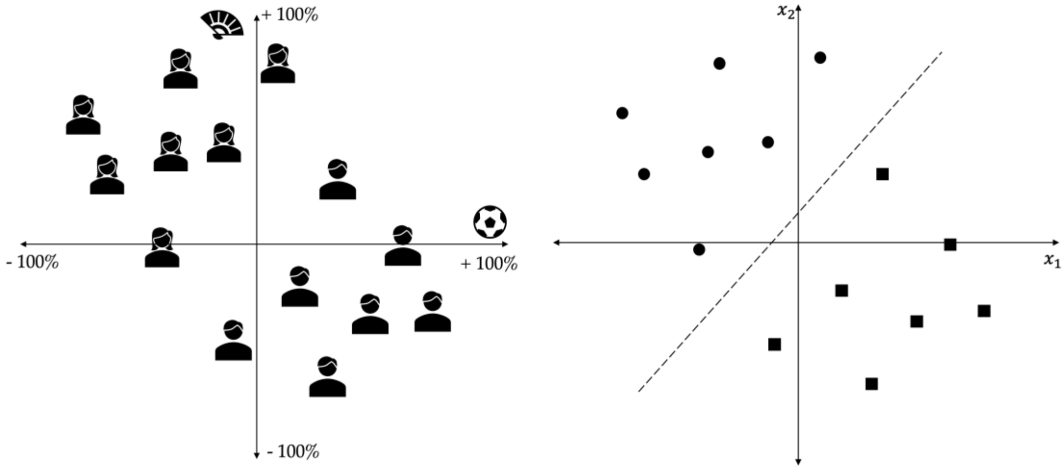
Gambar 2. Ilustrasi langkah-langkah *gradient descent* (tanda silang merah) untuk mencapai minimum pada fungsi biaya $J = \theta^2$.

3 Peran Konsep Kalkulus Dasar: dari Masalah Klasifikasi ke Pelatihan Jaringan Saraf Tiruan

Dalam berbagai referensi, jaringan saraf tiruan sering kali disebutkan sebagai konsep yang diilhami oleh cara kerja otak manusia [8,12], yaitu analog dengan saraf biologis yang berfungsi mengidentifikasi objek dan pola. Sejumlah literatur mencatat bahwa konsep jaringan saraf tiruan berakar pada makalah Warren McCulloch dan Walter Pitts pada tahun 1943, yang mencoba memformulasikan model matematis untuk menggambarkan fungsi sel-sel otak [6]. Penjelasan ini memang lebih mudah diterima dan dipahami, tetapi sering kali menutupi kenyataan bahwa jaringan saraf tiruan pada dasarnya adalah hasil konstruksi dari proses matematis yang terstruktur. Pada bagian ini, kita akan mengeksplorasi penggunaan konsep-konsep matematika yang telah diuraikan pada bab sebelumnya untuk membangun inti serta proses pelatihan jaringan saraf tiruan.

Kita mulai dengan analogi sederhana dan intuitif untuk membantu memahami harapan dan tujuan dari jaringan saraf tiruan. Bayangkan kita memiliki sekelompok orang yang ingin dikelompokkan ke dalam dua kelas: kelas pertama adalah laki-laki, dan kelas kedua adalah perempuan. Kedua kelompok ini tentunya memiliki banyak kesamaan minat, namun kita akan fokus pada dua aktivitas utama, yaitu berolahraga dan menari, sebagai cara sederhana untuk mengklasifikasikan mereka. Dengan membandingkan preferensi terhadap kedua aktivitas ini, kita dapat membuat asumsi yang membantu dalam mengelompokkan sebagian besar dari mereka.

Gambar 3.(a) mengilustrasikan situasi ini secara visual. Sumbu mendatar merepresentasikan tingkat kesukaan seseorang terhadap olahraga, sementara sumbu tegak menunjukkan seberapa besar minat seseorang terhadap menari. Dengan memetakan preferensi ini, kita dapat melihat pola yang memungkinkan klasifikasi berdasarkan kedua dimensi tersebut. Sebagai contoh, individu dengan skor tinggi pada olahraga sering kali lebih dikaitkan dengan laki-laki, sedangkan mereka yang menunjukkan skor tinggi pada menari lebih sering diasosiasikan dengan perempuan. Meskipun hal ini tidak selalu berlaku secara mutlak, ilustrasi ini cukup untuk mendemonstrasikan bagaimana proses klasifikasi dapat bekerja dalam konteks jaringan saraf tiruan.



(a) (b)

Gambar 3. Tingkat kesukaan seseorang terhadap olahraga dan menari pada sistem dua dimensi, dan (b) contoh garis pemisah antara kelompok laki-laki dan perempuan berdasarkan preferensi terhadap olahraga dan menari.

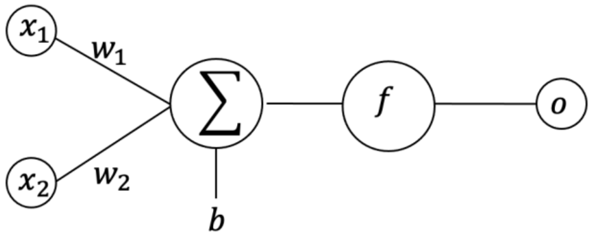
Salah satu cara sederhana untuk mengklasifikasikan kedua kelompok ini adalah dengan menggambar garis pemisah antara kedua kelas, seperti yang ditunjukkan pada garis putus-putus di Gambar 3.(b). Masalah berikutnya adalah bagaimana menemukan persamaan garis tersebut. Persamaan garis dapat dinyatakan ke dalam bentuk $w_1x_1 + w_2x_2 + b = 0$ dengan x_1 dan x_2 adalah koordinat yang menggambarkan preferensi terhadap olahraga dan menari. Sebagaimana kita ketahui, nilai w_1 dan w_2 memengaruhi kemiringan garis, sementara nilai b berkaitan dengan titik potong garis tersebut dengan sumbu tegak koordinat.

Untuk menemukan persamaan garis pemisah, kita misalkan titik-titik pasangan berurut (x_1, x_2) pada Gambar 3.(b) sebagai input, sementara *output*nya dihasilkan dari fungsi sesepenggal f berikut.

$$f(x_1, x_2) = \begin{cases} 1, \text{jika } w_1x_1 + w_2x_2 + b > 0 \\ 0, \text{jika } w_1x_1 + w_2x_2 + b < 0 \end{cases}. \tag{2}$$

Selanjutnya, w_1 dan w_2 disebut sebagai bobot yang dikalikan dengan x_1 dan x_2. *Output* dari fungsi f terdiri atas dua nilai yang menunjukkan kelas pengklasifikasian: 1 untuk laki-laki dan 0 untuk perempuan. Berdasarkan Gambar 3.(b), titik berbentuk bulatan mewakili titik-titik dengan target bernilai 0 (perempuan), sementara titik berbentuk persegi mewakili titik-titik dengan target bernilai 1 (laki-laki).

Fungsi f pada Persamaan (2) kita gunakan untuk membentuk model konseptual jaringan saraf tiruan. Merujuk pada Gambar 4, kita mulai dengan dua lingkaran yang mewakili input yang kemudian terhubung ke unit penjumlahan (sigma) yang bertugas memproses penjumlahan input setelah masing-masing dikalikan dengan bobotnya. Selain itu, kita juga memiliki bias, yang dinotasikan dengan b dan memiliki bobot bernilai 1.



Gambar 4. Model konseptual jaringan saraf tiruan berdasarkan klasifikasi dengan pencarian garis pemisah dua kelompok.

Hasil penjumlahan tersebut kemudian melewati proses pemfilteran oleh fungsi f, yang menghasilkan *output* 0 atau 1. Dengan demikian, *output* dari model ini hanya mencakup dua nilai. Fungsi f ini kita sebut sebagai fungsi label karena memiliki *output* yang setara dengan label atau kelas yang diinginkan. Fungsi yang berlaku sebagai filter ini sering disebut sebagai fungsi aktivasi karena ia mampu mengkonversi hasil penjumlahan menjadi nilai *output* yang sesuai dengan klasifikasi. Inilah model inti dari jaringan saraf tiruan, yang jika dilihat dari perspektif biologis, menggambarkan bagaimana input diterima oleh ujung saraf, diproses oleh neuron, dan setelah aktivasi, hasilnya diteruskan ke ujung saraf berikutnya.

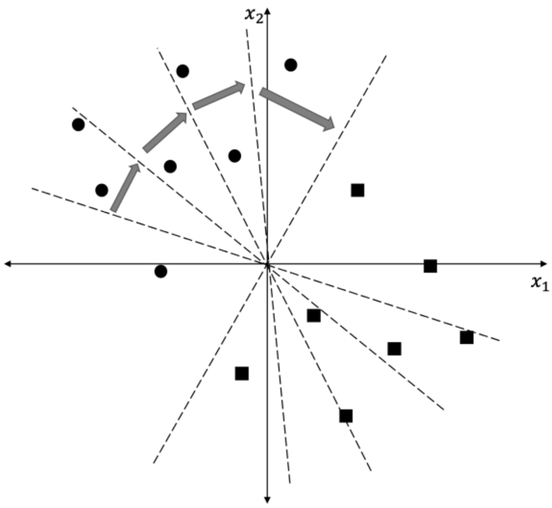
Kembali ke model yang kita lihat dari perspektif matematis, terdapat tak terhingga banyaknya kemungkinan untuk membentuk garis pemisah, dan tentu saja, kita ingin menemukan garis yang dapat mengklasifikasikan kelompok orang yang kita bahas dengan tepat ke dalam kategori laki-laki dan perempuan. Di sinilah kita akan mulai melakukan proses pembelajaran pada jaringan saraf sederhana yang telah kita buat pada Gambar 4. Dengan kata lain, berdasarkan informasi titik-titik yang akan kita klasifikasikan, kita perlu menemukan nilai w_1, w_2 dan b yang tepat. Dalam hal ini kita harus menentukan nilai w_1, w_2 dan b yang meminimumkan kesalahan (eror) antara *output* yang dihasilkan oleh fungsi f dan target yang dimiliki data.

Misalkan t^i adalah target yang diinginkan jaringan pada pembelajaran ke-i, dan o^i adalah *output* aktual yang dihasilkan oleh jaringan pada pembelajaran sampel ke-i.

Dengan n adalah banyaknya sampel dalam data, eror antara kedua nilai tersebut dapat dihitung menggunakan rumus *Mean Squared Error* (MSE) sebagai berikut.

$$C = \frac{1}{n}\sum_{i=1}^{n}\left(t^i - o^i\right)^2. \tag{3}$$

Dalam pembelajaran mesin, eror inilah yang umum disebut sebagai fungsi biaya. Selanjutnya, tujuan kita adalah untuk memperoleh nilai bobot dan bias yang dapat meminimumkan Persamaan (3). Proses ini dilakukan secara iteratif, dimulai dengan nilai awal untuk bobot dan bias, yang kemudian diperbarui menggunakan algoritma *gradient descent*. Diharapkan, pembaruan nilai bobot dan bias ini akan mengarah pada penemuan garis pemisah yang dapat memisahkan kedua kelompok dengan baik, seperti yang diilustrasikan pada Gambar 5.



Gambar 5. Ilustrasi penemuan garis pemisah antara dua kelompok melalui pembaruan nilai bobot dan bias.

Berdasarkan konsep algoritma *gradient descent*, kita dapat menggunakan Persamaan (1) untuk menuju nilai minimum eror atau minimum fungsi biaya. Langkah pertama adalah menghitung gradien fungsi biaya untuk n sampel yang diberikan. Fokus pertama kita adalah pembaruan bobot, dan setelah itu, kita akan melihat bagaimana pembaruan bias dilakukan. Misalkan bobot kita adalah $\mathbf{w} = [w_1,\ w_2]$. Untuk meminimumkan eror, kita turunkan Persamaan (3) terhadap bobot sebagai berikut.

$$\begin{aligned}\frac{\partial C}{\partial w_j} &= \frac{1}{n}\sum_{i=1}^{n} 2\left(t^i - o^i\right)\frac{\partial}{\partial w_j}\left(t^i - o^i\right)\\ &= \frac{2}{n}\sum_{i=1}^{n}\left(t^i - o^i\right)\frac{\partial}{\partial w_j}\left(t^i - \sum_{j=1}^{2} w_j x_j^i\right)\\ &= \tfrac{2}{n}\textstyle\sum_{i=1}^{n}\left(t^i - o^i\right)\left(-x_j^i\right).\end{aligned} \tag{4}$$

Selanjutnya, kita gunakan Persamaan (4) ini untuk memperbarui nilai bobot, yaitu $w_j := w_j + \alpha \Delta w_j$, dengan

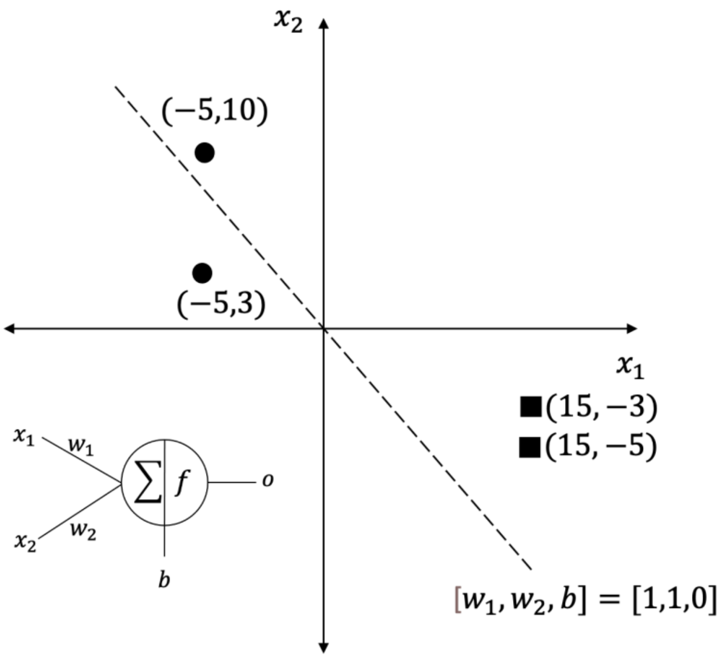
$$\Delta w_j = -\frac{\partial C}{\partial w_j} = -\frac{2}{n}\sum_{i=1}^{n}\left(t^i - o^i\right)\left(-x_j^i\right) = \frac{2}{n}\sum_{i=1}^{n}\left(t^i - o^i\right)\left(x_j^i\right). \tag{5}$$

α adalah *learning rate* yang menentukan seberapa besar langkah yang diambil dalam setiap iterasi.

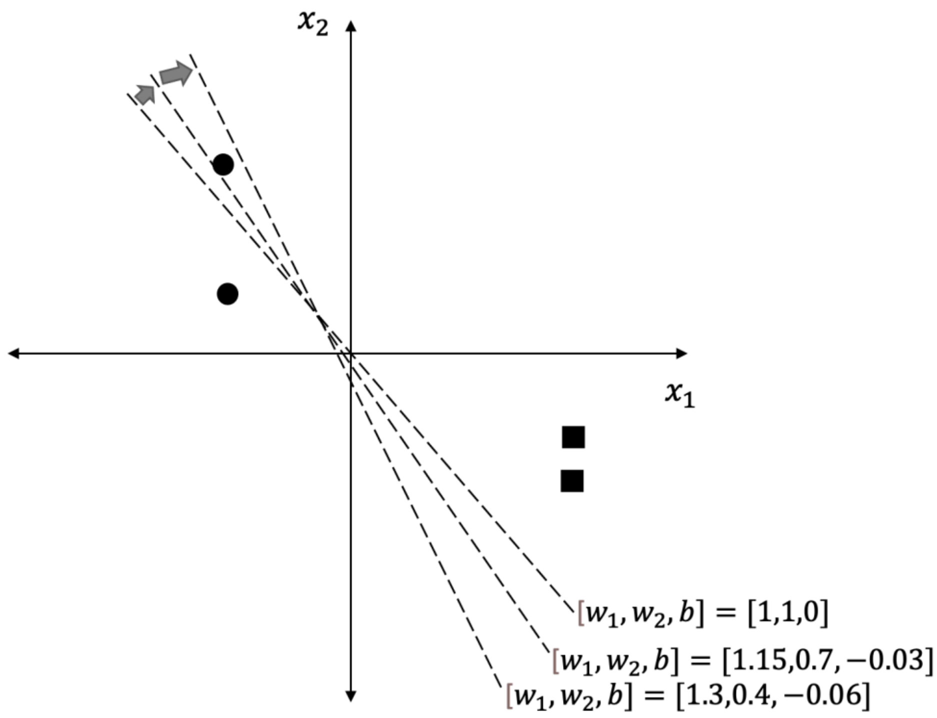
Untuk memperbarui nilai bias, kita dapat mengikuti langkah yang serupa dengan pembaruan bobot. Langkah pertama adalah menurunkan Persamaan (3) terhadap bias b. Dengan melakukan ini, kita memperoleh gradien terhadap bias yang memungkinkan kita untuk memperbarui nilai bias tersebut menggunakan rumus yang mirip dengan pembaruan bobot, yaitu $b := b + \alpha \Delta b$, dengan

$$\Delta b = -\frac{\partial C}{\partial b} = -\frac{2}{n}\sum_{i=1}^{n}\left(t^i - o^i\right)(-1) = \frac{2}{n}\sum_{i=1}^{n}\left(t^i - o^i\right). \tag{6}$$

Lebih detail lagi, kita terapkan penurunan rumus ini dalam bentuk contoh numerik untuk memahami bagaimana algoritma *gradient descent* menemukan nilai bobot koneksi dan bias yang optimal pada model jaringan saraf tiruan. Kita akan fokus pada empat titik sampel ($n = 4$), yaitu $(x_1^1, x_2^1) = (-5,10)$, $(x_1^2, x_2^2) = (15,-3)$, $(x_1^3, x_2^3) = (-5,3)$, dan $(x_1^4, x_2^4) = (15,-5)$. Untuk targetnya, kita memiliki $t^1 = t^3 = 0$ dan $t^2 = t^4 = 1$ sebagaimana diilustrasikan pada Gambar 6.(a).



(a)



(b)

Gambar 6. (a) Ilustrasi empat titik sampel dengan garis pemisah awal dan (b) proses pembaruan bobot serta bias untuk mendapatkan garis pemisah yang sesuai dengan target titik sampel.

Dengan keempat sampel tersebut dan mengambil nilai awal $[w_1, w_2, b] = [1,1,0]$, kita akan mulai menghitung gradien dari eror antara *output* aktual dan target, lalu melakukan pembaruan bobot dan bias untuk mencapai garis pemisah yang optimal. Proses ini mengarah pada iterasi perhitungan gradien yang dilakukan pada setiap sampel untuk menyesuaikan bobot dan bias.

Pada iterasi pertama, langkah pertama adalah menghitung *output* untuk masing-masing sampel menggunakan fungsi aktivasi f pada Persamaan (2). Diperoleh hasil sebagai berikut:

$$o^1 = f(w_1 x_1^1 + w_2 x_2^1 + b) = f((-5)(1) + (10)(1) + 0) = f(5) = 1,$$
$$o^2 = f(w_1 x_1^2 + w_2 x_2^2 + b) = f\big((15)(1) + (-3)(1) + 0\big) = f(12) = 1,$$
$$o^3 = f(w_1 x_1^3 + w_2 x_2^3 + b) = f((-5)(1) + (3)(1) + 0) = f(-2) = 0,$$
$$o^4 = f(w_1 x_1^4 + w_2 x_2^4 + b) = f\big((15)(1) + (-5)(1) + 0\big) = f(10) = 1.$$

Langkah kedua adalah menghitung gradien bobot dan bias menggunakan Persamaan (5) dan Persamaan (6).

$$\begin{aligned}\Delta w_1 &= \tfrac{2}{4}\textstyle\sum_{i=1}^{4}\big(t^i - o^i\big)\big(x_1^i\big)\\ &= \frac{1}{2}[(t^1 - o^1)(x_1^1) + (t^2 - o^2)(x_1^2) + (t^3 - o^3)(x_1^3) + (t^4 - o^4)(x_1^4)]\\ &= \tfrac{1}{2}[(0-1)(-5) + 0 + 0 + 0] = \tfrac{5}{2} = 2.5,\end{aligned}$$

$$\Delta w_2 = \frac{2}{4}\sum_{i=1}^{4}(t^i - o^i)(x_2^i)$$
$$= \frac{1}{2}[(t^1 - o^1)(x_2^1) + (t^2 - o^2)(x_2^2) + (t^3 - o^3)(x_2^3) + (t^4 - o^4)(x_2^4)]$$
$$= \frac{1}{2}[(0-1)(10) + 0 + 0 + 0] = -5,$$
$$\Delta b = \frac{2}{4}\sum_{i=1}^{4}(t^i - o^i) = \frac{1}{2}[(t^1 - o^1) + (t^2 - o^2) + (t^3 - o^3) + (t^4 - o^4)]$$
$$= \frac{1}{2}[-1 + 0 + 0 + 0] = -0.5.$$

Langkah terakhir adalah memperbarui bobot dan bias dengan memilih ukuran langkah $\alpha = 0.05$. Diperoleh,

$$w_1 := w_1 + \alpha(2.5) = 1 + (0.05)(2.5) = 1.125,$$
$$w_2 := w_2 + \alpha(-5) = 1 + (0.05)(-5) = 0.75,$$
$$b := b + \alpha(-0.5) = 0 + (0.05)(-0.5) = -0.025.$$

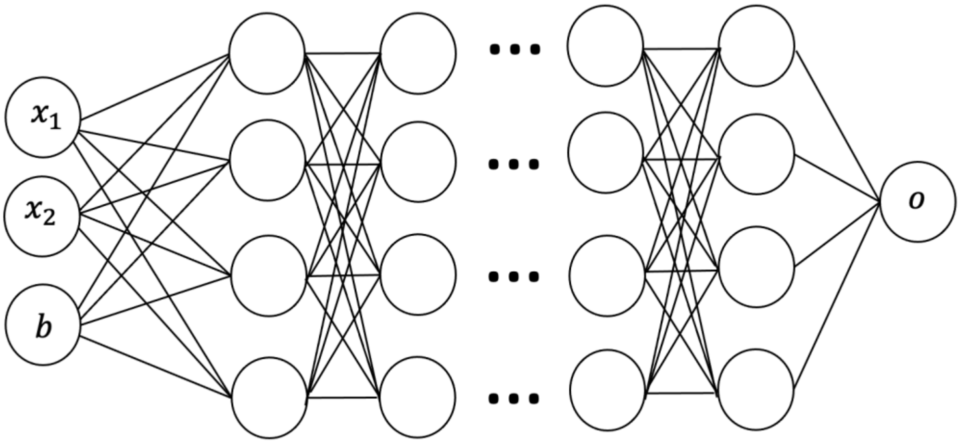
Nilai bobot dan bias yang diperbarui pada iterasi pertama digunakan sebagai dasar untuk iterasi kedua, dengan proses pembaruan dilakukan menggunakan langkah-langkah yang sama. Pada iterasi kedua dan ketiga, hasil perhitungan menunjukkan perubahan nilai bobot, bias, dan *output* secara bertahap hingga mendekati kondisi optimal. Semua hasil perhitungan dari iterasi pertama hingga ketiga dirangkum dalam Tabel 1 berikut.

Tabel 1. Hasil iterasi untuk pembaruan bobot dan bias berdasarkan algoritma *gradient descent*.

Iterasi	***Output*** $(\boldsymbol{o^1, o^2, o^3, o^4})$	**Gradien** $(\Delta \boldsymbol{w_1}, \Delta \boldsymbol{w_2}, \boldsymbol{b})$	**Pembaruan** $(\boldsymbol{w_1, w_2, b})$
1	$(1,1,0,1)$	$(2.5, -5, -0.5)$	$(1.125, 0.75, -0.025)$
2	$(1,1,0,1)$	$(2.5, -5, -0.5)$	$(1.25, 0.5, -0.05)$
3	$(0,1,0,1)$	$(0, 0, 0)$	$(1.25, 0.5, -0.05)$

Dari Tabel 1 terlihat bahwa iterasi kedua menghasilkan pembaruan bobot dan bias yang mendekati nilai optimal. Pada iterasi ketiga, gradien bernilai nol, menandakan bahwa proses pembaruan telah mencapai kondisi konvergen. Bobot dan bias yang diperoleh pada iterasi ketiga berhasil membuat *output* dari keempat titik sampel sesuai dengan nilai targetnya. Dengan demikian, kita telah memperoleh bobot dan bias yang mampu membentuk persamaan garis pemisah yang akurat untuk titik-titik sampel tersebut. Ilustrasi perubahan posisi garis pemisah dapat dilihat pada Gambar 6.(b).

Melalui contoh numerik ini serta penggunaan fungsi aktivasi sederhana untuk membentuk model jaringan saraf tiruan, kita telah memahami proses pelatihan jaringan yang terdiri atas satu neuron berdasarkan prinsip matematika. Pendekatan ini efektif untuk masalah yang bersifat *linearly separable*, yaitu ketika data dapat dipisahkan menggunakan garis linear. Namun, untuk masalah yang lebih kompleks, pengembangan jaringan diperlukan, seperti penambahan neuron dan lapisan yang menampung neuron (*hidden layer*) untuk menangani pola yang lebih rumit [7] sebagaimana diilustrasikan pada Gambar 7.



Gambar 7. Pengembangan jaringan saraf tiruan dengan penambahan neuron dan lapisan yang menampung neuron (*hidden layer*).

Lebih lanjut, fungsi aktivasi, fungsi biaya dan bahkan algoritma *gradient descent* dapat dimodifikasi sesuai kebutuhan. Semua pengembangan ini memerlukan perhitungan dan konsep matematika lanjutan yang tentunya lebih kompleks. Proses iterasi pun tidak lagi memungkinkan dilakukan secara tradisional melainkan membutuhkan bantuan komputer. Selain itu, ukuran sampel yang digunakan akan jauh lebih besar untuk memperoleh akurasi yang optimal, yang kini menjadi lebih mudah di era big data [4]. Inovasi semacam ini menjadi dasar bagi teknologi *deep learning*, yang menjadi salah satu fondasi utama pengembangan AI untuk menyelesaikan berbagai masalah praktis. Terlepas dari kompleksitasnya, kita dapat melihat bahwa semua konsep tersebut berakar pada matematika dasar, khususnya kalkulus, yang menjadi landasan penting dalam pembelajaran mesin.

4 Simpulan

Melalui kasus klasifikasi sederhana, telah ditunjukkan penggunaan konsep matematika dalam membentuk inti jaringan saraf tiruan. Materi dasar kalkulus, khususnya turunan, berperan penting dalam pelatihan jaringan ini. Contoh numerik dengan empat sampel data membantu kita memahami proses pembaruan bobot dan bias secara iteratif untuk meminimalkan eror hingga menghasilkan model yang sesuai dengan target. Hal ini menegaskan kuatnya keterkaitan antara pendekatan matematis dan perancangan model kecerdasan buatan. Pengembangan jaringan saraf untuk masalah yang lebih kompleks, seperti dalam *deep learning*, tetap berpijak pada prinsip yang sama. Penambahan *hidden layer*, penggunaan fungsi aktivasi yang lebih kompleks, dan modifikasi fungsi biaya serta proses optimasinya merupakan langkah lanjutan yang memerlukan perhitungan matematis lebih mendalam. Dengan demikian, penguasaan matematika dasar, khususnya kalkulus, menjadi landasan yang esensial bagi pengembangan teknologi AI yang semakin maju. Peran matematika sebagai fondasi ini mempertegas kontribusinya dalam membangun teknologi masa depan yang inovatif.

Daftar Pustaka

[1] Ahmad H, Wahyudi H. 2023. *Pembelajaran Matematika Era Digitalisasi*. Yogyakarta: Deepublish.

[2] Al-Khowarizmi S, Lubis AR 2023. *Artificial Intelligence*. Medan: Umsu Press.

[3] I. D. Id. 2021. *Machine Learning: Teori, Studi Kasus dan Implementasi Menggunakan Python*. Pekanbaru: Unri Press.

[4] Khan R, Usman M, Moinuddin M, 2024. The big data revolution: Leveraging vast information for competitive advantage. *Revista Espanola de Documentacion Cientifica*, 18(02): 65-94.

[5] Maesaroh S. 2024. *Pembelajaran Mesin dan Kecerdasan Buatan: Teori dan Aplikasi Praktis*. Serang: Sada Kurnia Pustaka.

[6] McCulloch WS, Pitts W. 1990. A logical calculus of the ideas immanent in nervous activity. *Bulletin of mathematical biology*, 52(1-2): 99-115.

[7] Mirjalili S. 2019. *Evolutionary algorithms and neural networks*. New York: Springer.

[8] Nurdiati S, Bukhari F, Najib MK, Hilmi K. 2022. Prediksi masa studi mahasiswa matematika IPB berdasarkan indeks prestasi kumulatif menggunakan jaringan syaraf tiruan. *MILANG Journal of Mathematics and Its Applications*, 18(1): 1-13.

[9] Ruder S. 2016. An overview of gradient descent optimization algorithms. Ruder.io. [diakses 2024 Des 1]. https://www.ruder.io/optimizing-gradient-descent/

[10] Stewart, J. 2003. *Kalkulus Jilid 1(4).* Jakarta: Erlangga.

[11] Varberg, Dale, Rigdon, Steven E, Purcell, Edwin J. 2003. *Kalkulus Jilid 1 Edisi 8* (Ed.8). Jakarta: Wiley.

[12] Zola F. 2018. Jaringan syaraf tiruan menggunakan algoritma backpropagation untuk memprediksi prestasi siswa. *Jurnal Teknologi Dan Open Source*. 1(1):58-72.