Para penemu kerap mencari inspirasi dari alam dan fenomena sehari-hari. Dulu pada tahun 1904 burung menginspirasi *Wright* bersaudara untuk menciptakan pesawat laik terbang perdana di dunia. Kini, desain futuristik kapal selam tenaga nuklir milik *Royal Navy Inggris*, contohnya, juga meniru bentuk paus, belut, dan ikan pari manta.

Demikian halnya dengan machine learning dan jaringan saraf tiruannya. Jaringan Saraf Tiruan atau *Artificial Neural Network* (ANN) adalah sebuah model machine learning yang terinspirasi dari *neuron/saraf* yang terdapat pada otak manusia.

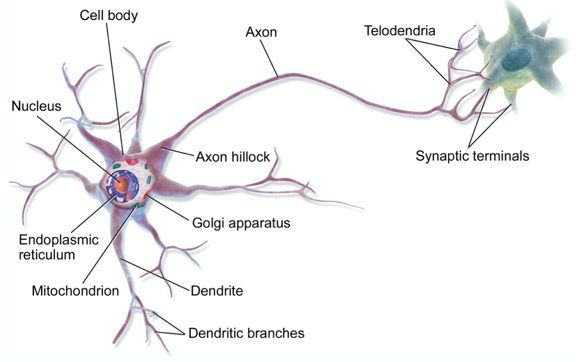
ANN merupakan salah satu model ML yang multiguna,*powerful*, dan memiliki skalabilitas tinggi. Dengan kelebihan tersebut ANN sangat ideal dipakai dalam menangani masalah ML yang sangat kompleks seperti mengklasifikasi miliaran gambar, mengenali ratusan bahasa dunia, merekomendasikan video ke ratusan juta pengguna, sampai belajar mengalahkan juara dunia permainan papan GO.

**Neuron dalam Otak**

Sebelum kita belajar mengenai saraf tiruan, kita akan mengenal lebih dahulu saraf biologis (neuron). *National Institute of Neurological Disorders and Stroke* [[18]](https://www.ninds.nih.gov/Disorders/patient-caregiver-education/life-and-death-neuron) dalam tulisannya yang berjudul “Brain Basics: The Life and Death of a Neuron” menyatakan bahwa neuron atau saraf adalah pembawa pesan/informasi. Mereka menggunakan impuls listrik dan sinyal kimiawi untuk mengirimkan informasi antara area otak yang berbeda, serta antara otak dan seluruh sistem saraf.

Sebuah saraf terdiri dari 3 (tiga) bagian utama, yaitu akson, dendrit, dan badan sel yang di dalamnya terdapat *nukleus*. Nukleus berisi materi genetik dan bertugas mengontrol seluruh aktivitas sel. Akson adalah cabang yang terlihat seperti ekor yang panjang, bertugas mengirimkan pesan dari sel. Panjang akson berkisar antara beberapa kali lebih panjang dari badan sel, sampai 10 ribu kali lebih panjang dari badan sel. Sedangkan dendrit adalah cabang-cabang pendek yang terlihat seperti cabang pohon, bertugas menerima pesan untuk sel.

Setiap ujung akson dari sebuah neuron terhubung dengan dendrit dari neuron lainnya. Neuron berkomunikasi satu sama lain dengan mengirimkan senyawa kimia yang disebut *neurotransmitter,* melintasi ruang kecil*(synapse)*antara akson dan dendrit neuron yang berdekatan. Ketika sebuah neuron mendapatkan rangsangan, neuron tersebut akan mengirim sinyal ke neuron lainnya. Seperti ketika kita tidak sengaja menyentuh panci yang panas, saraf di  tangan kita mengirim sinyal ke saraf lain sampai ke otak dan kita merespon dengan cepat. Pengiriman sinyal antar neuron terjadi sangat cepat yaitu hanya dalam beberapa milidetik.

[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

Cara kerja dari sebuah neuron sangatlah sederhana. Namun, neuron-neuron tersebut terorganisir dalam sebuah jaringan berisi miliaran neuron. Setiap neuron lalu terhubung dengan beberapa ribu neuron lainnya. Dengan jumlah yang luar biasa besar tersebut, banyak pekerjaan kompleks yang dapat diselesaikan. Sama dengan sebuah sarang semut yang sangat besar dan kompleks seperti gambar di bawah. Tak terbayang kan sekumpulan semut kecil yang membangunnya?

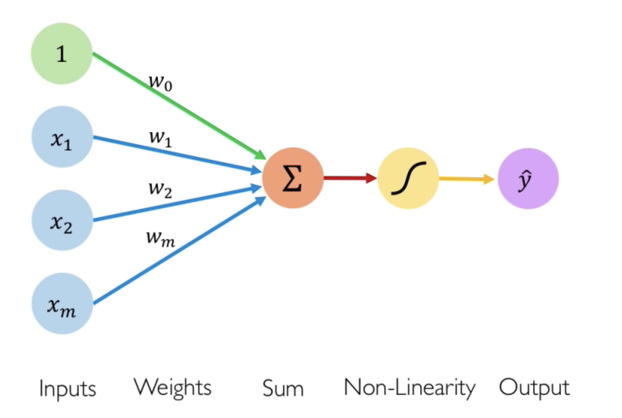
[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

**Perceptron**

Perceptron adalah komponen dasar pembangun jaringan saraf tiruan. Frank Rosenblatt dari Cornell Aeronautical Library adalah ilmuwan yang pertama kali menemukan perceptron pada tahun 1957 [19]. Perceptron pada jaringan saraf tiruan terinspirasi dari neuron pada jaringan saraf di otak manusia. Pada jaringan saraf tiruan, perceptron dan neuron merujuk pada hal yang sama.

Lantas bagaimana perceptron bekerja pada jaringan saraf tiruan?

Sebuah perceptron menerima masukan berupa bilangan numerik. Perceptron kemudian memproses masukan tersebut untuk menghasilkan sebuah keluaran. Agar lebih memahami cara kerja perceptron, kita akan menggunakan diagram di bawah.

[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

Sebuah perceptron terdiri dari 5 komponen yaitu:

1. Input (xi)
2. Bobot atau weights (Wi) dan bias (W0)
3. Penjumlahan atau sum (∑)
4. Fungsi aktivasi atau non linearity function (⎰)
5. Output (y)

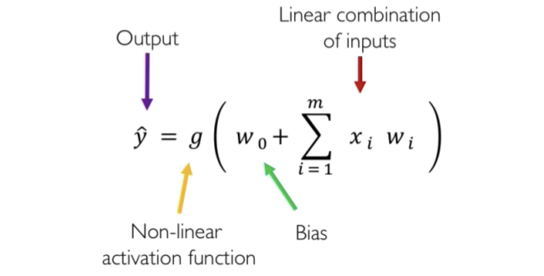
Berikut adalah proses yang menjelaskan bagaimana perceptron bekerja, yaitu:

* **Input**menerima masukan berupa angka-angka.
* Setiap input memiliki**bobot** masing-masing. Bobot adalah parameter yang akan dipelajari oleh sebuah perceptron dan menunjukkan kekuatan node tertentu.
* Selanjutnya adalah tahap **penjumlahan input.** Pada tahap ini setiap input akan dikalikan dengan bobotnya masing masing, lalu hasilnya akan ditambahkan dengan bias yang merupakan sebuah konstanta atau angka. Nilai bias memungkinkan Anda untuk mengubah kurva fungsi aktivasi ke atas atau ke bawah sehingga bisa lebih fleksibel dalam meminimalisasi eror. Penjelasan lebih lanjut tentang bias dapat Anda pelajari pada [tautan](https://towardsdatascience.com/why-we-need-bias-in-neural-networks-db8f7e07cb98)berikut.

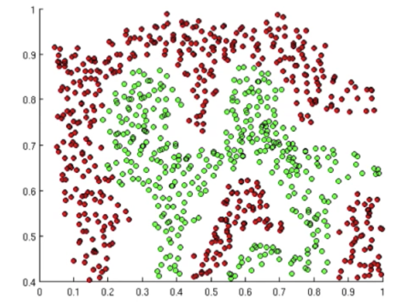
Hasil penjumlahan pada tahap ini biasanya disebut*weighted sum.*

* Langkah keempat, aplikasikan weighted sum pada fungsi aktivasi atau disebut juga **Non-Linearity Function**. Fungsi aktivasi digunakan untuk memetakan nilai yang dihasilkan menjadi nilai yang diperlukan, misalnya antara (0, 1) atau (-1, 1). Fungsi ini memungkinkan perseptron dapat menyesuaikan pola untuk data yang non linier. Penjelasan lebih lanjut tentang fungsi aktivasi akan diulas pada paragraf di bawah.
* Setelah semua langkah di atas, akhirnya kita memperoleh **output**, yaitu hasil dari perhitungan sebuah perceptron yang merupakan bilangan numerik.

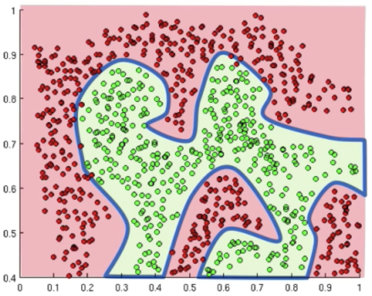
Fungsi matematis dari perceptron dapat kita lihat di bawah. Rumus di bawah merupakan notasi matematis yang menjelaskan proses yang kita bahas sebelumnya. **Keluaran (ŷ)** dari perceptron merupakan **bias (W0),** ditambah dengan jumlah setiap **input (Xi)** yang dikali dengan bobot masing-masing **(Wi)** sehingga menghasilkan weighted sum, kemudian dimasukkan ke dalam**fungsi aktivasi (g).**

[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

Fungsi aktivasi pada perceptron bertugas untuk membuat jaringan saraf mampu menyesuaikan pola pada data non linier. Seperti yang sudah pernah dibahas sebelumnya, mayoritas data yang terdapat di dunia nyata adalah data non linier seperti di bawah.

[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

Fungsi aktivasi lah yang memungkinkan jaringan saraf dapat mengenali pola non-linier seperti di bawah. Tanpa fungsi aktivasi, jaringan saraf hanya bisa mengenali pola linier seperti garis pada regresi linier.

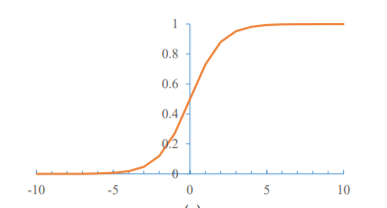
[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

Ada 3 fungsi aktivasi yang paling umum yaitu *sigmoid function, hyperbolic tangent,*dan *rectified linear unit*(ReLU). Berikut adalah penjelasannya.

**Sigmoid atau Logistic Function.**

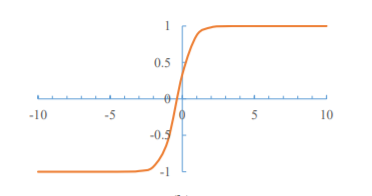
Fungsi ini berada di antara nilai 0 hingga 1 sehingga biasanya digunakan untuk memprediksi model probabilitas yang outputnya ada di kisaran 0 dan 1. Dengan kemiringan yang halus *(smooth gradient)*memungkinkan *Gradient Descent* (algoritma pengoptimalan yang mampu menemukan solusi optimal untuk berbagai masalah) berprogres pada setiap langkahnya.

Selain itu, fungsi sigmoid memberikan nilai prediksi yang lebih jelas. Dari gambar di bawah [20], perhatikan bahwa nilai X di bawah -2 atau di atas 2 cenderung memberikan nilai prediksi yang sangat dekat dengan 0 atau 1.

[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

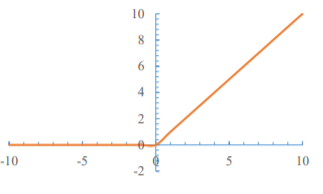
**Hyperbolic Tangent (tanh)**

Sama seperti fungsi sigmoid, fungsi tanh berbentuk S, kontinu, dan dapat dibedakan. Perbedaannya adalah nilai keluarannya berkisar dari -1 hingga 1. Rentang tersebut cenderung membuat keluaran setiap lapisan kurang lebih berpusat di sekitar 0 pada awal pelatihan sehingga dapat membantu mempercepat konvergensi [4]. Sifat*zero centered*ini juga membuat fungsi ini lebih mudah dalam memodelkan masukan yang memiliki nilai sangat negatif, netral, dan sangat positif.

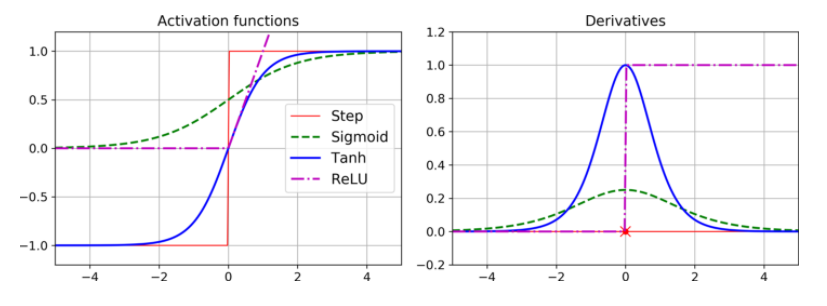
[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

**Rectified Linear Unit (ReLU)**

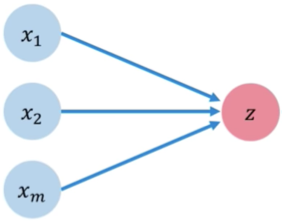
Fungsi ReLU bersifat kontinu meski kemiringannya berubah secara tiba-tiba dan nilai turunannya bernilai 0 pada z < 0. Akan tetapi, fungsi ini bekerja dengan sangat baik dan membuat jaringan bekerja secara efisien sehingga mempercepat waktu komputasi. Karena hal inilah, fungsi aktivasi ini sering digunakan sebagai fungsi aktivasi *default*pada jaringan saraf tiruan/ANN.

[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

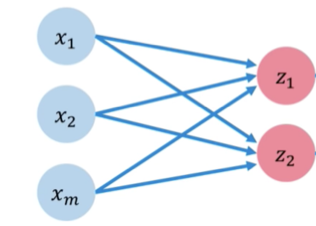
Beberapa fungsi aktivasi dan turunannya ditunjukkan pada gambar berikut.

[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

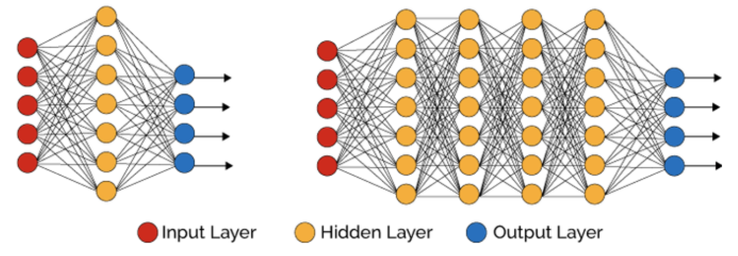
Selanjutnya mari kita kembali pada pembahasan mengenai perceptron. Kita akan menggunakan diagram yang lebih sederhana untuk mengilustrasikan perceptron. Pada diagram di bawah, **x1, x2,**dan**x3**adalah**input,** dan **z** adalah**perceptron.** Bias dan bobot tidak masuk dalam ilustrasi agar diagram terlihat lebih sederhana.

[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

Pada gambar di bawah terdapat 2 buah perceptron yaitu **Z1**dan **Z2.**Karena koneksi tiap perceptron dan tiap input sangat padat, jadi dinamakan dengan *dense layer.* Dense layer adalah sebuah lapisan yang terdiri dari 2 perceptron atau lebih.

[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

Selanjutnya kita akan membahas tentang hidden layer. Sebuah hidden layer adalah dense layer yang berada di antara input layer dan output layer. Pada ilustrasi di bawah, jaringan saraf di sebelah kiri memiliki 1 hidden layer dan jaringan saraf di sebelah kanan memiliki 4 buah hidden layer.

[](https://www.dicoding.com/academies/184/tutorials/8502?from=8497)

Dalam sebuah jaringan saraf tiruan, input layer dan output layer harus selalu ada, namun untuk hidden layer bisa ada beberapa atau tidak sama sekali. Hidden layer dan output layer sama-sama merupakan sebuah layer yang memiliki beberapa perceptron. Sedangkan input layer adalah sebuah layer yang hanya menampung angka-angka.

Hidden layer diberikan nama hidden karena sifatnya yang tersembunyi. Pada sebuah sistem jaringan saraf, input dan output layer merupakan lapisan yang dapat kita amati, sementara hidden layer, tidak.

Pada sebuah jaringan saraf tiruan, semakin banyak jumlah hidden layer dalam sistem, semakin lama jaringan saraf tersebut memproduksi hasil, namun juga semakin kompleks masalah yang dapat diselesaikan.