What are neural networks?

(requires only high-school maths)

甄景贤 (King-Yin Yan)

General.Intelligence@Gmail.com

The 3 main approaches in artificial intelligence are:

- logic
- neural networks
- evolution

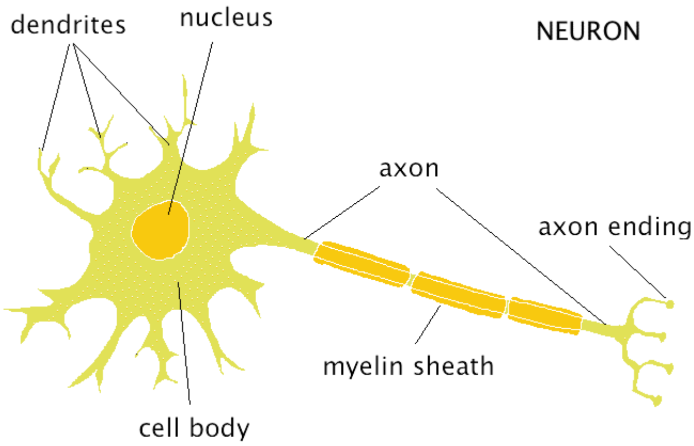
Neural networks is a special kind of **statistical learning**, that operates on "points" in a vector space.

Deep learning is the hottest technique in current AI research. "Deep" simply means "many layers of neural networks".

Biological neurons

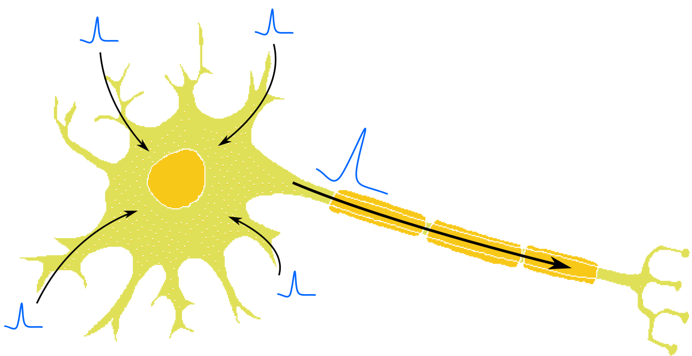
Let's refresh some high-school biology ☺

This is a biological neuron:



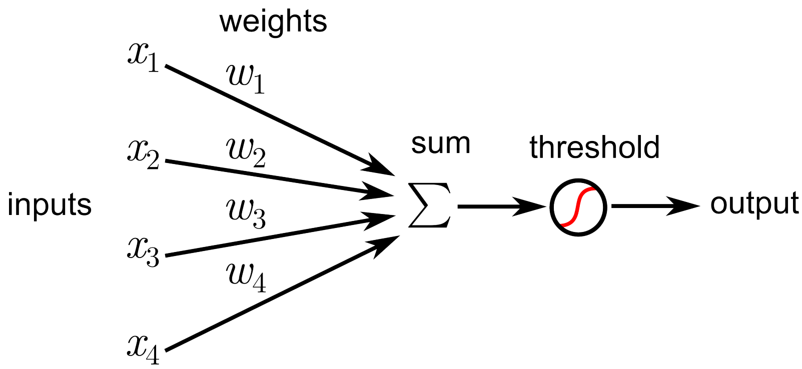
(1)

Dendrites **collect** electrical signals; When the **total** of these signals exceed a certain **threshold**, the neuron **fires** an electric impulse, sending to another neuron via the **axon**:



(2)

Mathematically, this can be greatly simplified to such a **model**:



(3)

That means: each input value is **weighted** and then summed together, and then passes through a function for output.

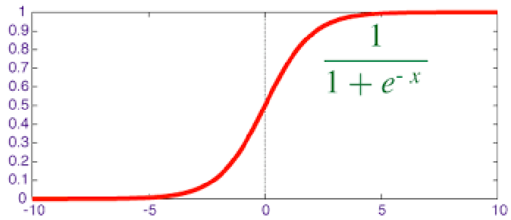
In formula:

$$\boxed{\text{output}}\ y = [\ \sum_i (w_i\ x_i)\ ] \tag{4}$$

where = sigmoid function, is defined by:

$$(x) = \frac{1}{1+e^{-x}} \tag{5}$$

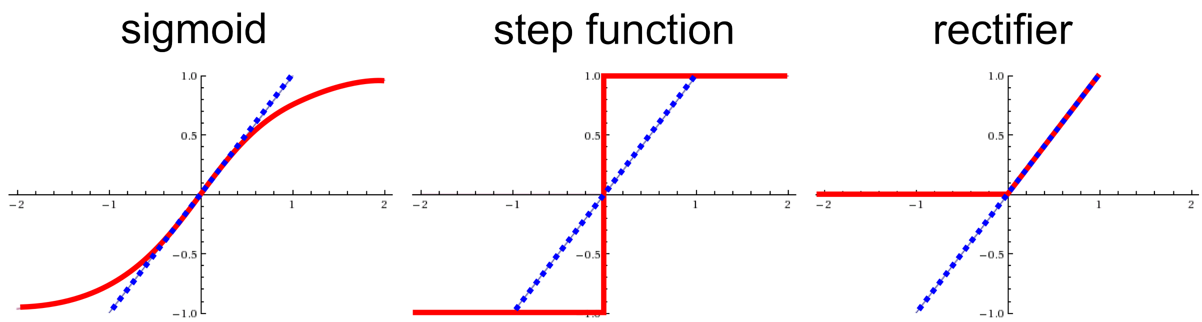
Its shape is like this:



(6)

On the left it is 0 = nothing (no signals), on the right it is 1 = "yes".

These functions can all play the role of the "threshold":



(7)

Fun fact #1

The surface of the neuron is covered with sodium-potassium (Na^+/K^+) channels, that use ATP (adenosine triphosphate, the energy source in the cell) to "pump" ions into the cell with a 3 Na^+ : 2 K^+ ratio, creating a voltage difference. When the voltage exceeds the threshold, certain ion channels open, the built-up voltage is released to create an "action potential". This phenomenon can be described using differential equations, that is the famous **Hodgkin-Huxley** equation, and its simplified version, the **FitzHugh-Nagumo** equation.

A very special feature about the action potential is that it is "**all-or-nothing**", ie, if the input is below threshold, the output signal would be flat (zero). Why is it like this? That is because the human brain evolved from primitive **multi-cellular** organisms (like the jellyfish) whose cells gradually learned to use electric signals to communicate. They operated in an environment like a pool of water, in which there is a lot of **noise**. Even now the human brain is like a pool of liquid, and the body is in constant motion, resulting in **heat noise**. In order to operate within such noise, there must be a mechanism to **suppress** the noise; This is the reason for all-or-nothing. That is to say, human consciousness has **finite** information content, similar to a digital computer, and is not mysterious.

Fun fact #2

The neuron's **cell membrane** is a **lipid bi-layer**, made up of fats and cholesterol. The function of cholesterol is to make the membrane structurally stable, therefore every cell needs cholesterol. The "wires" in the brain are all made of cell membranes, so the brain is basically made up of fats and cholesterol. In particular, the pig's brain which is a kind of Chinese food, has the highest cholesterol content of all foods, many times more than eggs!

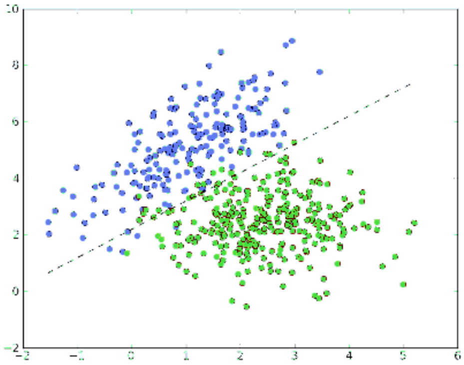
The **myelin sheath**, unique to vertebrates, is like the plastic insulation of electric wires, whose effect is to speed up electrical transmission. The octopus is an invertebrate, that's why it needs a big head to achieve the same level intelligence as vertebrate animals of comparable brain size.

Fun fact #3

When the nerve signal reaches the **synapse**, it no longer uses electrical transmission, but switches to chemical transmission with **neuro-transmitter** molecules. There are many types of neuro-transmitters, such as **serotonin** and **dopamine**, often mentioned in anti-depression drugs. But the most common neuro-transmitter is **glutamate**, the main signaling molecule for the nervous systems of all animals. Plants lack a nervous system, therefore glutamate is not found in plants. Humans like to eat meat, so we evolved a taste for meat, especially the taste for glutamate. A Japanese scientist discovered a substance in sea-weed, which when added to food mimics the taste of meat. Actually this substance is just glutamate, or MSG (mono-sodium glutamate). So MSG is not harmful to humans, it's just that we may not get a balanced nutrition if we eat MSG often instead of real meat.

1 neuron – geometric interpretation

As I explained in other tutorials the goal of **machine learning** is usually to **classify** certain "points" in a space:



(8)

For example, in **machine vision**, a picture can have millions of **pixels**, each pixel being a dimension, its **color** is the **coordinate value** on this dimension. The entire space is the space of **all images**, with each **point** representing an image. The dimensionality of such spaces is very high (the dimension is the number of pixels per image). We often use 2 or 3 dimensions for explaining things, but the reader should use their imagination for higher dimensions.

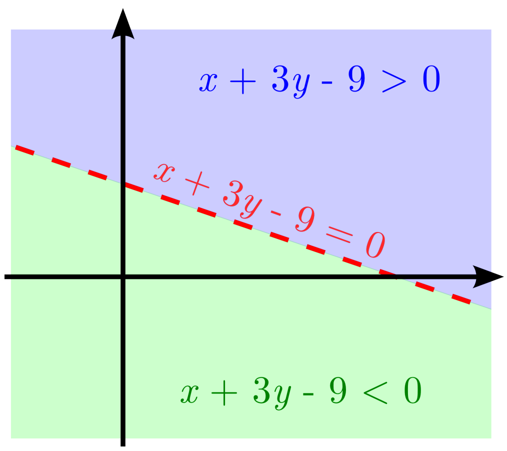
We know from high-school maths, the equation for a **straight line** is:

constants

$$ax + by + c = 0 \tag{9}$$

variables

Its **geometric interpretation** is like this:



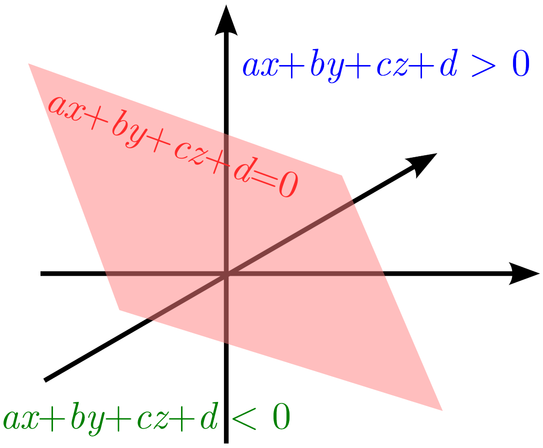
(10)

For points on the line, the equation is $= 0$. The line **cuts** the space into 2 halves: one side is > 0, the other < 0.

Generalizing to 3-dimensions, we have the equation for a **plane**:

$$ax + by + cz + d = 0 \tag{11}$$

It also **cuts** the space into 2 halves, one side > 0, the other < 0:



(12)

For arbitrary n-dimensions, with each point denoted as $\boldsymbol{x} = (x_1, x_2, ..., x_n)$, a **hyperplane** cuts the space into 2 halves, its equation is:

$$a_1 x_1 + a_2 x_2 + ... + a_n x_n + a_0 = 0 \tag{13}$$

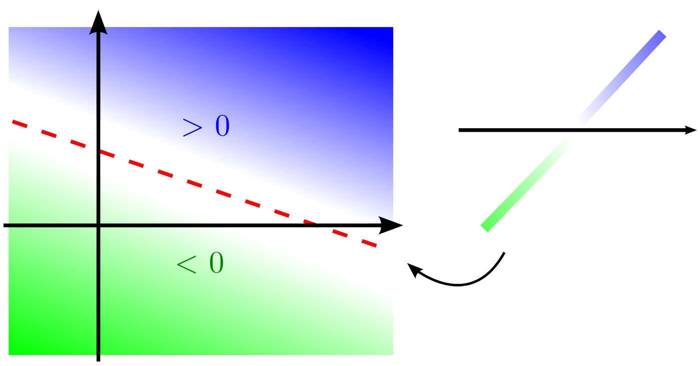
> Note: What is the dimension of a hyper-plane? In 2-D space, it is a line (1-D), in 3-D space, it is a plane (2-D); In general, in n-D space a hyper-plane is an $(n-1)$-dimension object, $(n-1)$ is also called **co-dimension 1**, meaning that the ambient space is of dimension n, and equation (13) reduces the **degrees of freedom** by 1, so the object **constrained** by this equation has $n - 1$ degrees of freedom.

Now we can see some resemblance between the hyper-plane and the neuron, as the neuron is a **linear combination** before passing to the :

$$\boxed{\text{output}}\ y = \quad [\overbrace{\sum_i (w_i\ x_i)}^{\text{linear combination}}] \tag{14}$$

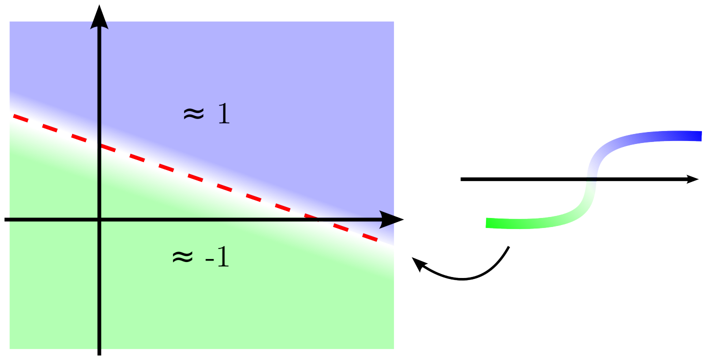
That is to say: <u>Each neuron forms a hyper-plane, that cuts the space into 2 halves.</u>

What if is applied? Without , the 2 halves are > 0 and < 0; Now if we see colors as "intensity", the intensity changes gradually: (the figure on the right shows the side view, as in 3-D)



(15)

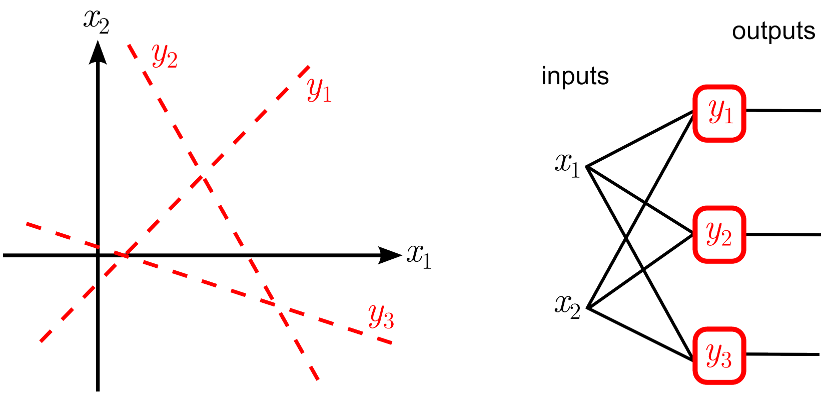
When is applied, 1 represents “yes” and 0 represents “no”, so the contrast between the 2 sides is enhanced, ie, more **polarized**, or binary:



(16)

¿ 1 neuron

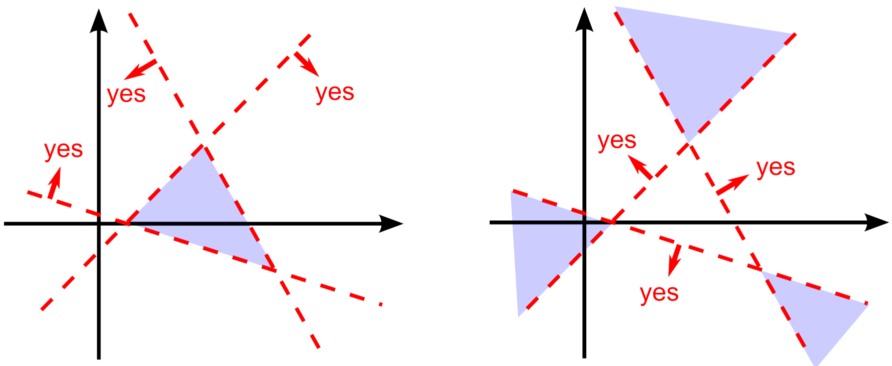
If there are > 1 neurons (on the same layer), eg. 3:



(17)

Note: The coordinates are (x_1, x_2) = input. Output is (y_1, y_2, y_3), represented by 3 dotted lines. The **network topology** of this **1 layer** of neural network is as shown on the right (The $\sum$ and for each neuron are not shown).

Each neuron may choose one side to be “yes”. The **conjunctions** of these choices may form various shapes, eg:



(18)

Also notice that in the right figure, a number of disjoint regions (separated in space) are possible.

Obviously, we could use neurons to separate and classify points in space, thus achieving the goal of **machine learning**.

1 layer of neurons

In this part we need more **linear algebra**, which I will explain in another tutorial.

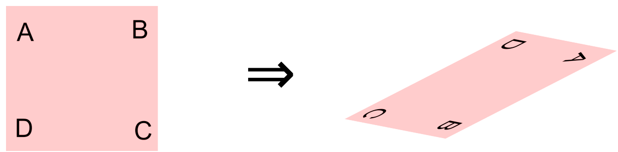
The mathematical form of 1 layer of neural network is:

$$\boxed{\text{output}}\ \boldsymbol{y} = \sigma[W\boldsymbol{x}] \tag{19}$$

where W is a matrix, ie. a **linear transformation**; σ is a **non-linear transformation**.

Matrices are a compact way of writing **systems of linear equations**. 1 neuron $= \sigma \circ \boxed{\text{linear combination}}$, each linear combination is a **row** in a matrix.

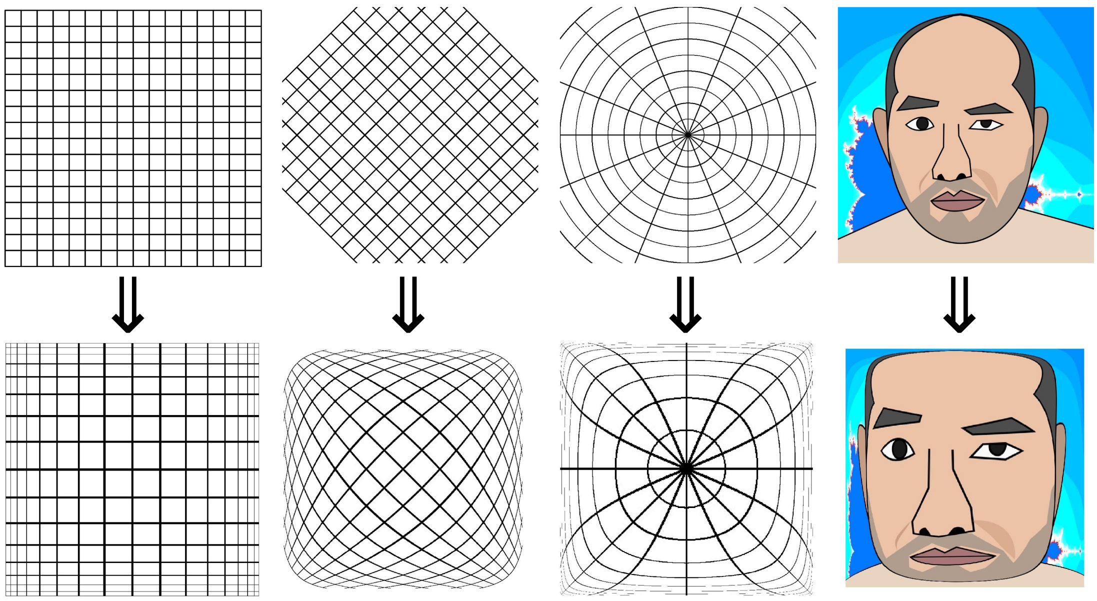
Every matrix is a representation for a **linear transformation**:



(20)

Linear transformations can "skew" a square into a parallelogram, and also include **rotations** and **translations**. Straight lines are preserved as straight lines, hence its name.

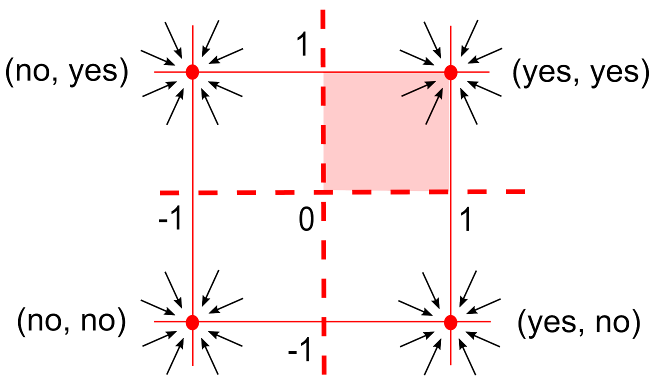
Next we need to know what σ does: (below is the sigmoid transform of both x and y coordinates)



(21)

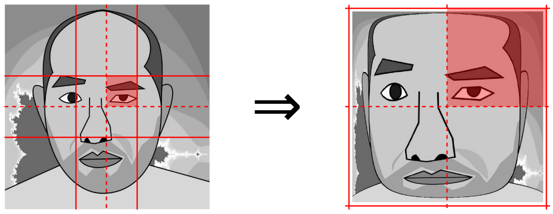
σ can be seen as "stretching" the square to its 4 corners, so I take on the "square-faced" look.

We say that these vertices are **attractors**:



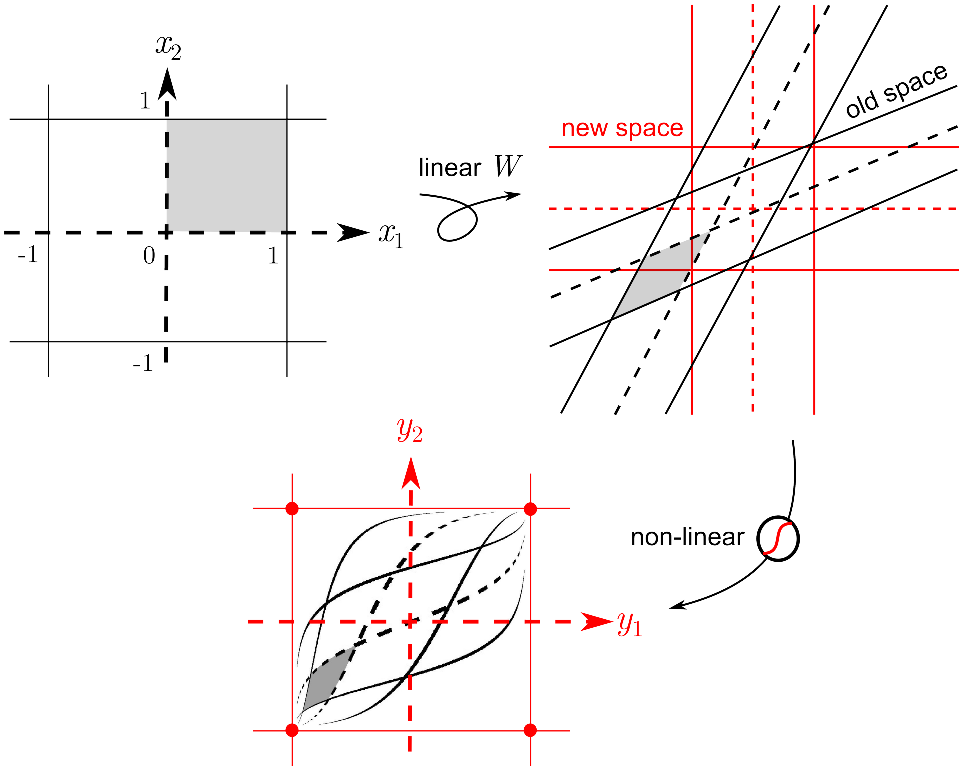
(22)

Note: During the $\textcircled{\int}$ transform, the image is compressed from the **infinite** unbounded space to the $[-1, +1]^2$ square:



(23)

Here is the transformation performed by **1 layer** of neural network, decomposed into the W and $\textcircled{\int}$ parts:



(24)

Multi-layer neural network

This is simply the repetition of single layers:

$$\boxed{\text{output}}\ \boldsymbol{y} = \textcircled{\int}\ \overset{1}{W}\ \ \textcircled{\int}\ \overset{2}{W}\ \ .....\ \textcircled{\int}\ \overset{L}{W}\ \ \boldsymbol{x} \tag{25}$$

L = total number of layers.

— The end —